AD-A068 351   MAGNAVOX GOVERNMENT AND INDUSTRIAL ELECTRONICS CO TO--ETC F/G 17/2.1
                PHASE DISTORTION STUDY.(U)
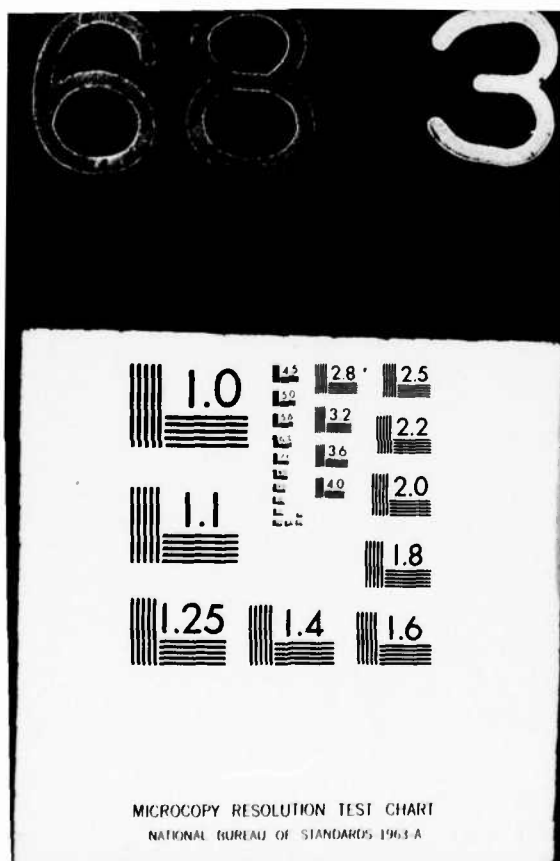   AUG 76   C R CAHN, C L MAY   DAAB07-76-C-0001
UNCLASSIFIED   R-8486   NL

1 OF 4
AD
AO   351

MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

MRL TECHNICAL INFORMATION NOTICE

All entries must be completed prior to printing. If a particular block is
not applicable, place N/A (not applicable) in the appropriate block.

DISTRIBUTION DATE _____

AUTHOR C. R. CAHN C. L. MAY	CONTRIBUTORS R. L. CRUPI	MRL REFERENCE NO. R-5456
		DOCUMENT DATE August 20, 1976

TITLE

PHASE DISTORTION STUDY
FINAL REPORT
Contract No. DAAB07-76-C-0001

SUBJECT/KEY WORDS
Phase distortion, AM/PM, Limiting, QPSK, Equalization, Coding,
Duobinary, FDMA, TDMA, Computer simulation

GOVERNMENT CLASS, AND MRL CONTROL NO. UNCLASSIFIED	TYPE OF INFORMATION TECHNICAL	NO. OF PAGES 378
MAGNAVOX CLASS N/A		NO. OF ILLUST. 155

ABSTRACT/CONCLUSIONS
This is the Final Report of a 9 month study of channel distortions on
digital data transmission over the DSCS. Computer simulations and
supporting analysis were developed for evaluating the degradation for
QPSK/FDMA and QPSK/TDMA. Emphasis was placed on the use of
adaptive equalization to correct severe phase distortion and alleviate
intersymbol interference due to narrow bandwidth filtering. Several
equalizer configurations and control strategies were developed for inter-
facing with present modems. Error correction coding was shown to
significantly reduce degradation due to adjacent channel interference and
intermodulation produced by channel saturation and AM/PM conversion.

The study also investigated techniques for improved bandwidth utili-
zation. For FDMA, the spacing between adjacent signals is reduced,
and for TDMA, a higher signaling rate is enabled by narrowband equal-
ized filters. With conventional QPSK, data transmission at 2 bps/Hz
becomes feasible. Duobinary demodulation of QPSK achieves 3 bps/Hz
at no increase in E_b/N_o through use of a maximum-likelihood demodu-
lation technique and appropriate channel equalization. Performance is
shown to be good even for saturating channels, and the rate-1/2 coding
gain is comparable to that for the memoryless channel.

BY CUTTING OUT THIS RECTANGLE AND FOLDING ON THE CENTER LINE,
THE ABOVE INFORMATION CAN BE FITTED INTO A STANDARD CARD FILE.

INFORMATION PREPARED FOR: _____ U. S. Army Satellite Communications Agency, Ft. Monmouth,

New Jersey

APPROVED BY: PROGRAM MANAGER _____ C. L. May _____ DATE: 8/20/76

DEPARTMENT MANAGER _____ C. R. Cahn _____ DATE: 8/20/76

PHASE DISTORTION STUDY
FINAL REPORT
Contract No. DAAB07-76-C-0001

Prepared for:

U. S. Army Satellite Communications Agency
Fort Monmouth, New Jersey 07703

Prepared by:

Magnavox Government and Industrial Electronics Company
Advanced Products Division
2829 Maricopa Street
Torrance, California 90503

Approved:

C. R. Cahn, Associate Director Advanced Programs
C. L. May, Program Manager

R-5456
August 20, 1976

TABLE OF CONTENTS

IV. ERROR CORRECTION CODING

V. EQUALIZERS

APPENDICES

LIST OF ILLUSTRATIONS

Figure No.

LIST OF TABLES

SECTION I
STUDY SURVEY AND KEY RESULTS

This report contains the results of the Phase Distortion Study performed for the U. S. Army Satellite Communications Agency by the Magnavox Government and Industrial Electronics Company. The original objective of this 9 month study was the evaluation of the combined phase distortion effects for digital data transmission over the Defense Satellite Communication System (DSCS). Phase distortion problems appear to have been minimal in the DSCS in the past, but with the increased use of quadriphase modulation for high rate data transmission, the management of solutions to phase distortion is expected to require increasing attention.

The phase linearity requirements of the various elements of the DSCS earth/satellite system have been specified individually at values believed to contribute negligible distortion to the planned signal types. However, the overall degradation of phase distortion effects on the system performance has not previously been quantified, and the overall effects of cumulative distortion were not precisely known.

In addition, although some means are currently provided for the improvement of channel phase linearity in earth terminals by simple switchable phase equalizer structures, on-line adaptive equalization is likely to be a future requirement. The best method for adjusting such equalizers and the types of links for which adaptive equalization is necessary has not been established. Therefore, this study had the basis of establishing the ground rules for approaches to solutions for these problems which arise in the current FDMA system operation and also in future high burst rate TDMA operation.

The approach followed in this study is to evaluate performance by means of a time domain computer simulation which represents the overall BPSK/QPSK digital communication system. Computer program SIMA was developed for this purpose, and includes simulations of system elements such as filters, phase distortion characteristics, saturating elements, modems, error correction, and adaptive equalizers. Using these simulations, the DSCS system phase characteristics are studied and the degradation resulting from the various components of phase distortion

evaluated parametrically. Various forms of adaptive equalizers are investigated for use in the earth terminals, and a preliminary design approach for an adjustable IF equalizer is developed in some detail. A number of error control strategies suitable for on-line operation are studied for equalizer weight adjustment and their operation evaluated by simulation. Based on the results obtained, a systematic procedure for specifying tolerances on components of the overall phase distortion is developed.

Although emphasis was originally placed on adaptive equalization of excessive phase distortion in the channel, it was found that benefits can be derived from equalization of a narrow channel bandwidth so as to improve the bandwidth utilization of the channel (i.e., increase the data rate in the available bandwidth). Bandwidth requirements for both FDMA and TDMA operation are evaluated for a variety of linear and saturating bandlimited channels. For FDMA operation, the Ground Mobile Forces scenario was simulated to determine the feasibility of spacing adjacent accessing signals by frequency separations approaching the data rate. For TDMA operation, since both the ground terminals and the satellite will probably operate in the limiting mode to maximize link power and to prevent the need for power control of the accesses, the effects of multiple hard limiters and narrowband filtering of QPSK and SQPSK is evaluated.

Also, rate-1/2 error correction convolutional coding/Viterbi decoding for reduction of the degradation to FDMA and TDMA due to phase distortion and amplitude saturation is studied.

The shifting of study emphasis towards improved bandwidth utilization led to a preliminary investigation of the duobinary signaling technique. The results are very promising. Using an adaptive equalizer to create the channel response necessary for duobinary operation, it is found that the QPSK data rate can be increased to 3 bps/Hz, without significant degradation in required E_b/N_o when a relatively simple maximum likelihood demodulator is employed. Utilization of rate-1/2 convolutional coding over the duobinary channel is also studied, along with the structure of the applicable Viterbi decoder.

This Final Report is divided into ten major sections, each pertaining to a specific topic addressed in the study. In addition, there are five appendices, which contain supplementary information and supporting analyses developed during the investigation plus a description of the SIMA software.

The following results, based on this study, are of major importance to the DSCS system. Additional conclusions, recommendations, and areas requiring further investigation are given in Section 10.

- The phase distortion characteristics of the DSCS terminals and satellites will not significantly degrade performance of the 10 Mbps QPSK/BPSK modems.

- Error correction coding (which reduces the data rate for a fixed symbol rate) reduces degradation for channels with phase distortion, saturation, or adjacent channel interference unless the distortion is so severe that the hard decision error rate is extremely high.

- The decision-directed adaptation of an IF equalizer can be controlled from bit decision samples out of the modem. Even simpler, it is found that good equalization is achieved with control to minimize the alpha-flunk output of the QPSK/BPSK modem. Sequential dither control of the equalizer tap weights is a feasible adjustment strategy. With error corrected data, control based on the decoder output has been found unnecessary (and would be incompatible with the use of alpha flunk).

- Adaptive equalization of phase distortion in the nonsaturating channel eliminates almost all degradation produced by the distortion.

- For FDMA operation, sharp cutoff filters are required in both the transmitter and receiver to permit close spacings of the adjacent signals, so as to achieve the maximum bandwidth utilization. The use of linear phase filters with proper amplitude shaping and a bandpass equal to the bit rate for BPSK (or half the total bit rate for QPSK) improves system performance when an integrate-and-dump detector is used. This permits QPSK operation (uncoded) at 2 bps/Hz with minor degradation and, in the linear channel, allows FDMA signal spacing by as little as 0.55 times the total data rate (for uncoded QPSK). With either the Phase II or the Phase III satellite TWT, use of amplitude-shaped filters permit signal separations as close as 0.6 times the total data rate (for uncoded QPSK).

o   For FDMA operation at lower error rates with uncoded data, an optimum TWT output backoff exists at which the total system loss (defined as the output backoff from saturation in dB plus the E_b/N_o degradation) is minimized. For FDMA operation at higher error rates, or with rate-1/2 error correction, the TWT should be operated at saturation.

o   Error correction coding of the FDMA signals and use of amplitude-shaped filters reduces degradation in the saturating channel and permits signal spacings of 1.1 times the data rate. (Rate-1/2 coding and QPSK is assumed.) Degradation is also reduced by error correction coding when the earth terminal transmitters hard limit, and feasible signal separation is 1.3 times the data rate.

o   Duobinary signaling can increase the bandwidth utilization by 50 percent (to 3 bps/Hz) with no penalty in E_b/N_o over QPSK with conventional demodulation, assuming use of the Viterbi algorithm for maximum-likelihood demodulation of the duobinary and an equalizer in the receiver to produce the necessary duobinary channel response. Duobinary signaling is compatible with a saturating channel, with a small degradation penalty in E_b/N_o.

SECTION II
BANDWIDTH AND PHASE DISTORTION CHARACTERISTICS
OF DSCS TERMINALS AND SATELLITES

The following paragraphs examine the bandwidth and phase distortion characteristics of the various DSCS terminals and of the DSCS Phase II satellite and prospective Phase III satellite, as available in existing documentation (either by specification or from actual measurements).

Amplitude distortion is not discussed here because the ripple introduced by the system elements is typically small in the passband and does not contribute significantly to degradation. To see why the effect of amplitude ripple is small, the concept of paired echoes is convenient. Thus, let the amplitude response as a function of frequency have a small cosine ripple, so that the channel transfer function in the frequency domain is

$$Y(j\omega) = 1 + a \cos(\omega T_e)$$

$$= 1 + \frac{a}{2} e^{j\omega T_e} + \frac{a}{2} e^{-j\omega T_e} \tag{1}$$

By Fourier transform theory, this distortion produces a pair of symmetrical echoes of amplitude a/2, respectively, leading and lagging by T_e. Combining the echoes from the various bits, intersymbol interference of maximum amplitude a is produced. From (1), note that a = 0.1, giving only a modest amount of intersymbol interference, corresponds to a relatively severe gain ripple of $\pm.85$ dB.

The concept of paired echoes applies to sine ripple-type phase distortion. also. Let the channel transfer function be

$$Y(j\omega) = 1 + ja \sin(\omega T_e)$$

$$= 1 + \frac{a}{2} e^{j\omega T_e} - \frac{a}{2} e^{-j\omega T_e} \tag{2}$$

Now, the channel has only a slight amplitude ripple, but has an approximately* sinusoidal phase ripple of $\pm\tan^{-1}(a)$. The ripple produces an antisymmetric pair of echoes, which can combine to give intersymbol interference of maximum amplitude a. Note that a = 0.1 yields a phase ripple of $\pm 5.7^{\circ}$ (or ± 0.1 radian), and this causes as much performance degradation to BPSK or QPSK as a gain ripple of $\pm .85$ dB.

2.1   AN/MSC-46 TERMINAL

Phase distortion data is available only for the high power klystron transmitter of the AN/MSC-46. The low power TWT transmitter is of much wider bandwidth and presumably contributes negligible phase distortion over the bandwidth of interest. The receiver is also very wideband and presumably contributes negligible phase distortion.

Two examples of phase distortion in the high power amplifier (HPA) are given in Figures 2-1 and 2-2. They are obtained by numerically integrating a measured group delay characteristic** according to the definition of group delay

$$\tau = d\varphi/d\omega \qquad\qquad (3)$$

where φ is in radians, ω is in rad/sec, and τ is in seconds.

Figure 2-1 displays predominately parabolic phase distortion centered at approximately 7.93 GHz. The distortion is 45° at a frequency offset of +60 MHz or -45 MHz from 7.93 GHz. Superimposed is a phase distortion ripple of a few degrees. Figure 2-2 shows much less parabolic distortion, but again the ripple is only a few degrees.

---

*Sinusoidal phase ripple without amplitude distortion produces multiple paired echoes, but (2) gives the dominant first-order echoes when a is small.

**Note, the phase distortion curves available for the MSC-46 have an ordinate scale not in degrees, but rather in nanoseconds multiplied by frequency in units of 1.25 MHz. The ordinate scale must be multiplied by the factor $1.25 \times 10^{-3} \times 2\pi \times 57.3 = 0.45$ to convert to degrees. This has been done to obtain Figures 2-1 and 2-2.
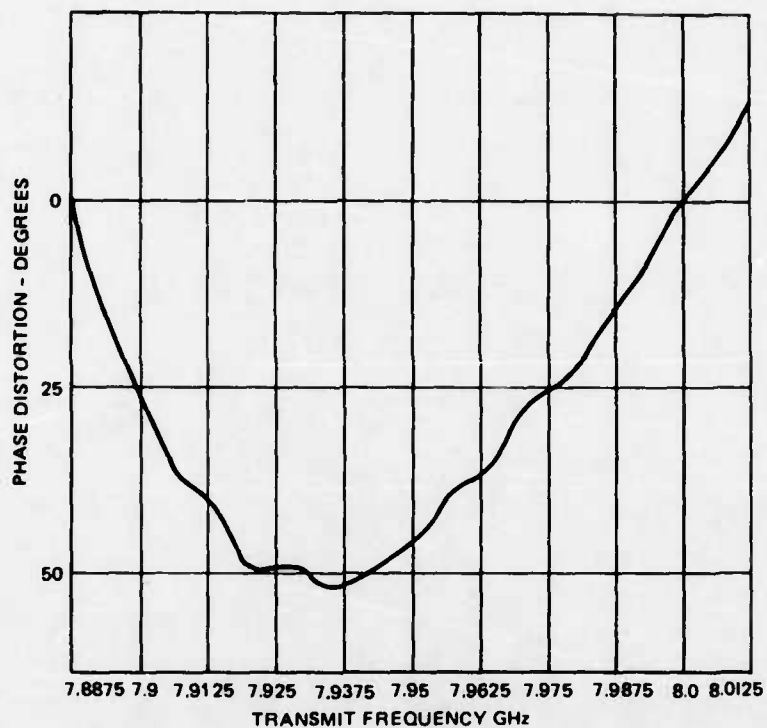
Figure 2-1. Integrated Group Delay Curve - Band 1 of AN/MSC-46 Transmitter
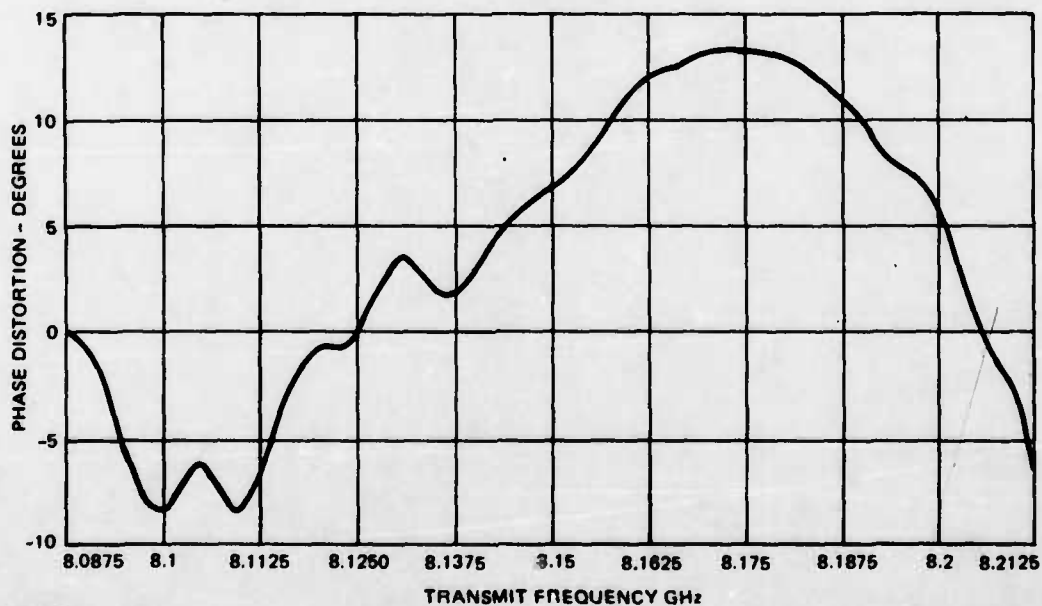
Figure 2-2. Integrated Group Delay Curve - Band 4 of AN/MSC-46 Transmitter

2.2 COMTECH CONVERTER

The Comtech Converter, Frequency CV-3084A/MSC-46 is used to up-convert any signal in the IF band of 70 MHz ±20 MHz into the frequency range 7.9 GHz to 8.4 GHz, for amplification and transmission. To change the RF, only the frequency of the L.O. supplied to the second up-conversion mixer need be changed. The up-converter is equalized for the quadratic and cubic components of phase distortion.* Figure 2-3 plots the phase distortion, obtained by numerically integrating the measured delay distortion (with delay equalization). The unequalized ripple is roughly ±1.5°.

Provision is made to insert a mop-up equalizer following the input to the up-converter.

Y-axis: PHASE DISTORTION - DEGREES (4.0, 3.0, 2.0, 1.0, 0, -1.0, -2.0, -3.0)

X-axis: IF FREQUENCY - MHz (50 52 54 56 58 60 62 64 66 68 70 72 74 76 78 80 82 84 86 88 90)

776-3832
UNCLASSIFIED
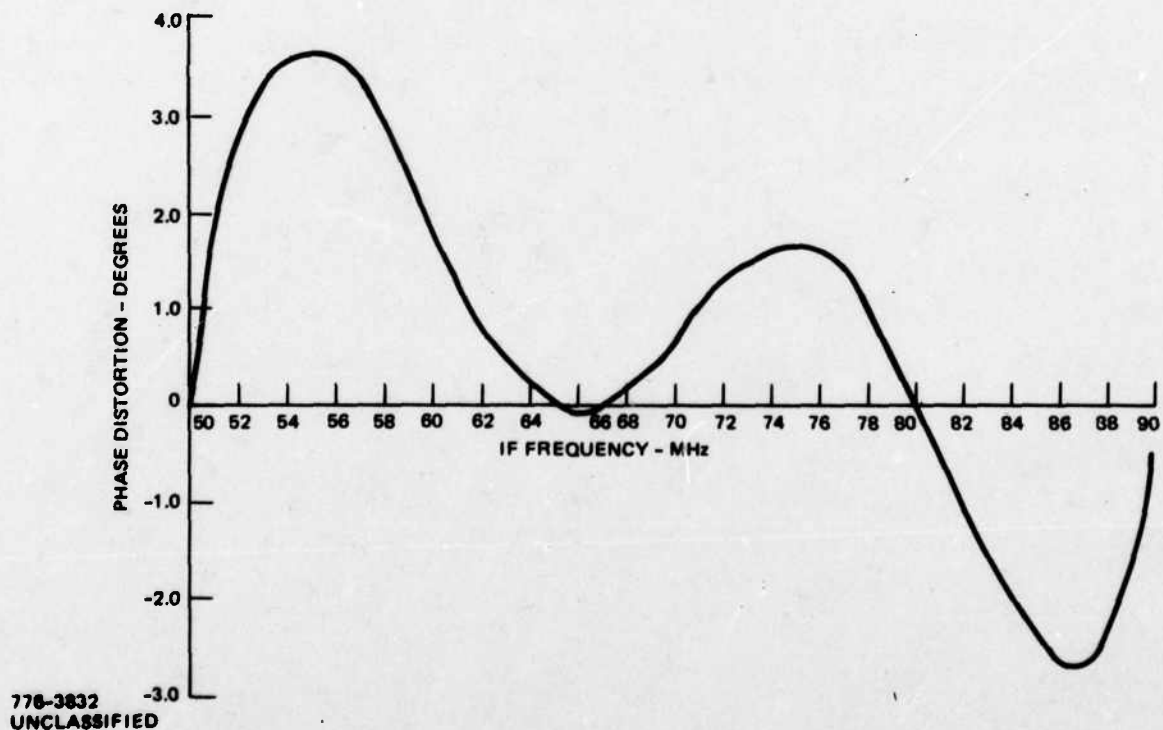
Figure 2-3. Measured Phase Distortion of COMTECH Up Converter
(Converter, Frequency CV-3084A/MSC-46)

*In terms of group delay, this means equalization of linear and quadratic delay distortion.

The Comtech Converter, Frequency CV-3085A/MSC-46 is capable of converting a signal in the frequency band 7.25 GHz to 7.75 GHz to IF's of 700 MHz and 70 MHz. The bandwidth is ±20 MHz at either IF output port. The RF can be changed by the Receive Frequency Selector; which tunes the frequency synthesizer to the correct frequency.

The down converter is equalized in the 70 MHz portion. The equalizer compensates for the parabolic and cubic phase distortion components. Figure 2-4 plots the phase distortion for an RF input of 7.5 GHz, obtained by numerically integrating the measured delay distortion (with delay equalization). The unequalized ripple is approximately ±1.5°.

Provision is made to insert a mop-up equalizer in the 70 MHz IF.

A terminal employing the COMTECH up-converter and down-converter is specified to have 40 MHz bandwidth at the 70 MHz IF. The bandpass at 70 MHz is set by a filter in the 700 MHz IF. Figure 2-5 is the measured response of this filter indicating a bandpass exceeding the 40 MHz requirement. The measured amplitude response is down by approximately 0.5 dB over a bandpass of 52 MHz. The bandwidth at the 700 MHz IF is 125 MHz.
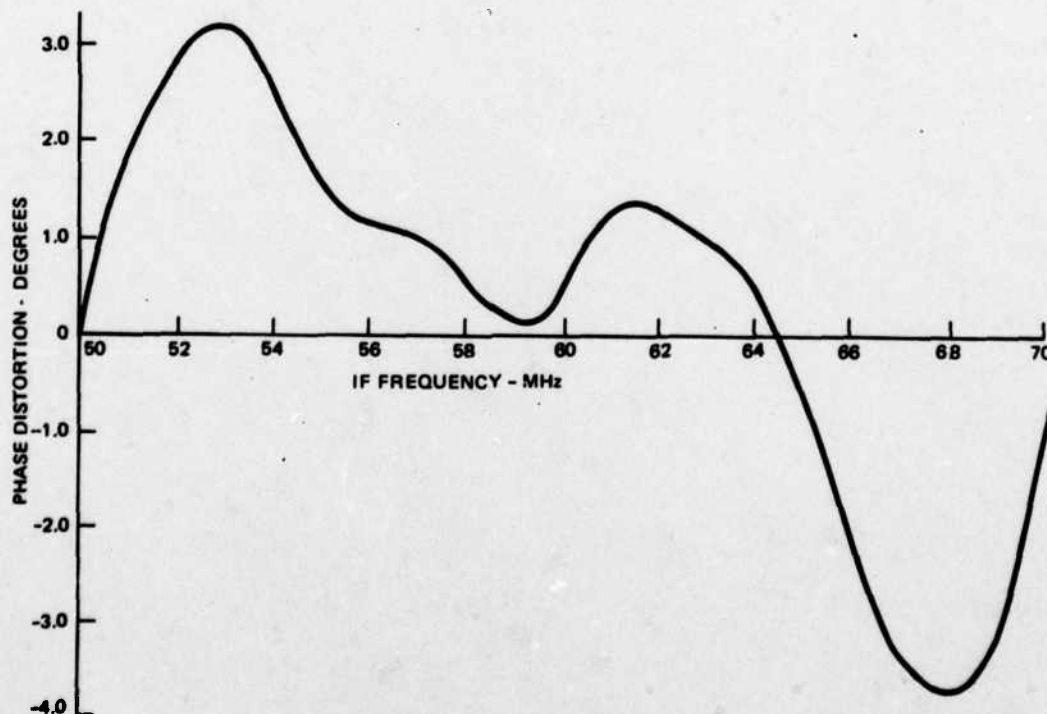
Figure 2-4. Measured Phase Distortion of COMTECH Down Converter at 7.5 GHz RF (Converter, Frequency CV-3085A/MSC-46)

2-5
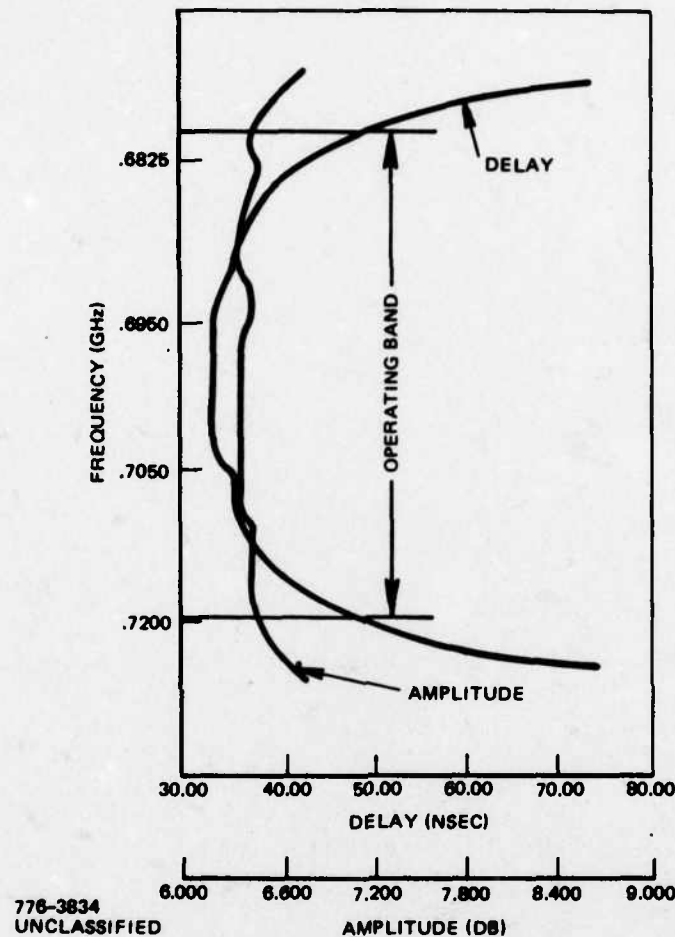
Figure 2-5. Converter 70 MHz IF Response (Set By 700 MHz Filter)

2.3   COMTECH VARIABLE 70 MHz GROUP DELAY EQUALIZER MODEL
          GDE-705

          The Comtech Model GDE-705 Group Delay Equalizer enables mop-up
equalization with variable linear and parabolic delay (parabolic and cubic phase).
The equalizer can provide up to ±15 nsec of linear delay variation over the 40 MHz
bandwidth.   Up to 15 nsec of parabolic delay variation over this bandwidth is also
provided.

          The parabolic phase correction provided by the maximum linear delay
variation of 15 nsec is

$$\phi_{degrees} \;=\; .0675 \; f^2_{MHz} \tag{4}$$

where f_{MHz} is the frequency offset in MHz from center. At f_{MHz} = 20 MHz, the maximum parabolic phase correction is 27°.

The cubic phase correction provided by the maximum parabolic delay variation of 15 nsec is

$$\varphi_{degrees} = .0045 \, f_{MHz}^3 \tag{5}$$

At f_{MHz} = 20 MHz, the maximum cubic phase correction is 36°.

2.4   AN/TSC-54 TERMINAL

Phase distortion of the AN/TSC-54 transmitter is measured from the 70 MHz input of the up converter to the output of the high power amplifier (HPA). The group delay is measured and integrated to give the phase distortion characteristic over a 40 MHz bandwidth.

Figure 2-6 shows an example where the phase shift is predominately parabolic, with a small ripple component in addition. As seen, the parabolic distortion is approximately 0.3 radians, or 17° over the 40 MHz band. The ripple is roughly $\pm 2^O$. Figure 2-7 shows predominately a ripple type phase distortion, of magnitude ± 0.1 radian, or $\pm 6^O$.

Phase distortion of the AN/TSC-54 receiver is measured from the RF input to the 70 MHz IF output of the down converter. Figure 2-8 shows an example where the phase distortion is primarily parabolic. The distortion is 0.36 rad, or 20° over the 40 MHz bandwidth. Figure 2-9 is an example where the phase distortion is roughly sinusoidal with one cycle of variation over the 40 MHz. The distortion is $\pm .07$ rad, or $\pm 4^O$.

2.5   AN/FSC-78 HEAVY TERMINAL

The AN/FSC-78 Heavy Terminal (HT), has wideband RF elements which contribute a negligible phase distortion. The only data available is the phase shift of the high pass filter following the high power amplifier. Figure 2-10 is a plot of the phase shift, which is primarily quadratic over the 500 MHz band. The phase distortion is very small over any 40 MHz bandwidth. The Fleet Broadcast Terminal AN/FSC-79 and Medium Terminal AN/MSC-61 have the same electronics and the same results apply to these as given for the AN/FSC-78.

Thus, the primary contribution to phase distortion in the AN/FSC-78 is from the Comtech up converter in the transmitter and the down converter in the receiver, as plotted above in Figures 2-3 and 2-4.
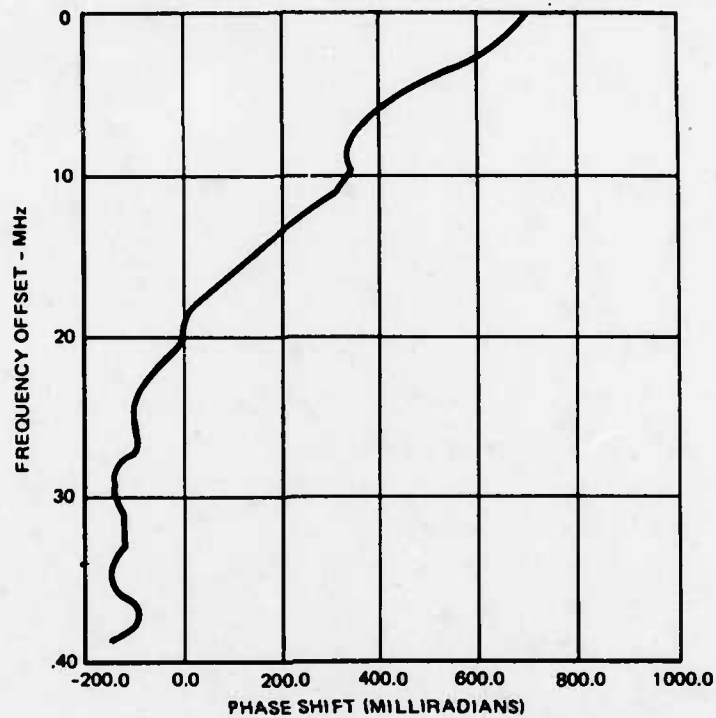
776-3835
UNCLASSIFIED

Figure 2-6. Transmit Phase Distortion - AN/TSC-54
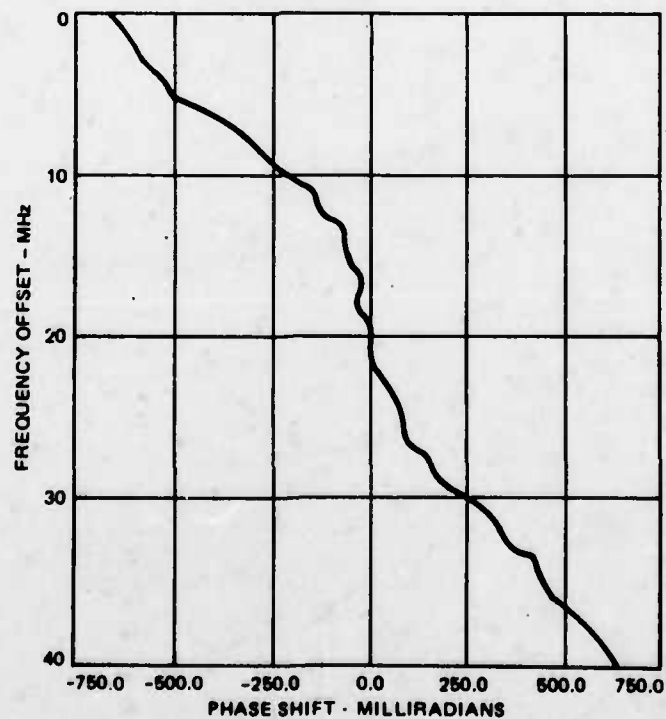
776-3836
UNCLASSIFIED

Figure 2-7. Transmit Phase Distortion - AN/TSC-54
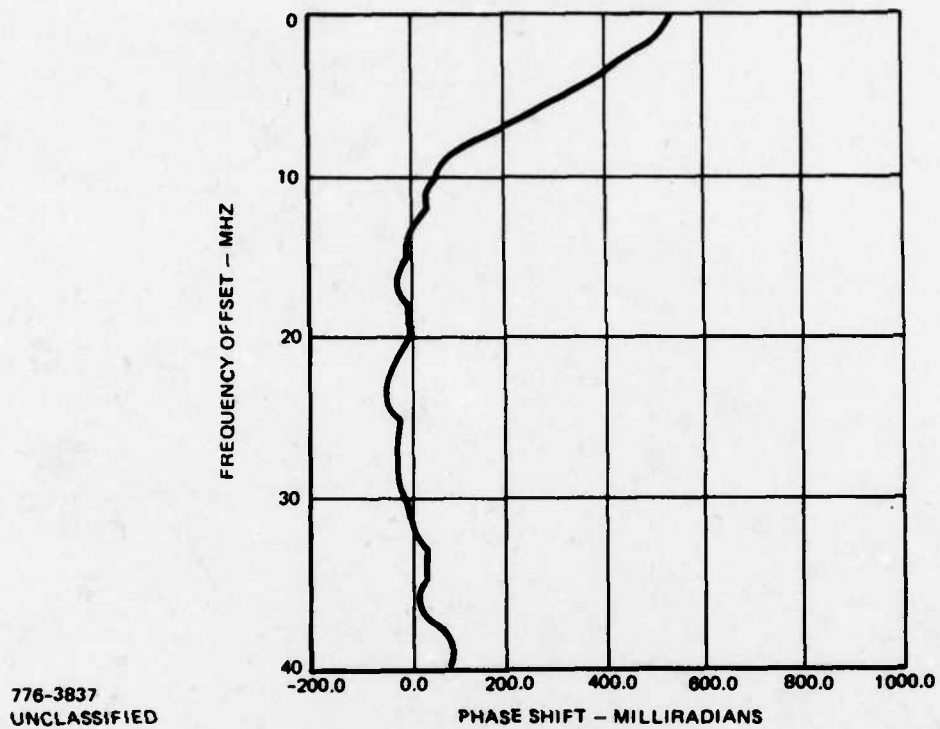
Figure 2-8. Receive Phase Distortion - AN/TSC-54
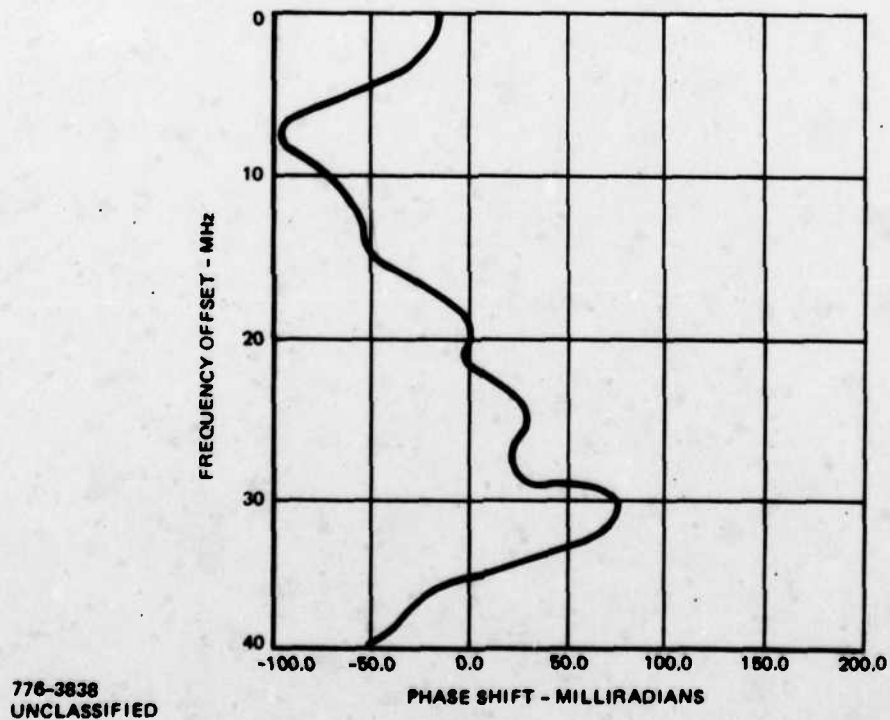
Figure 2-9. Receive Phase Distortion - AN/TSC-54

2-9
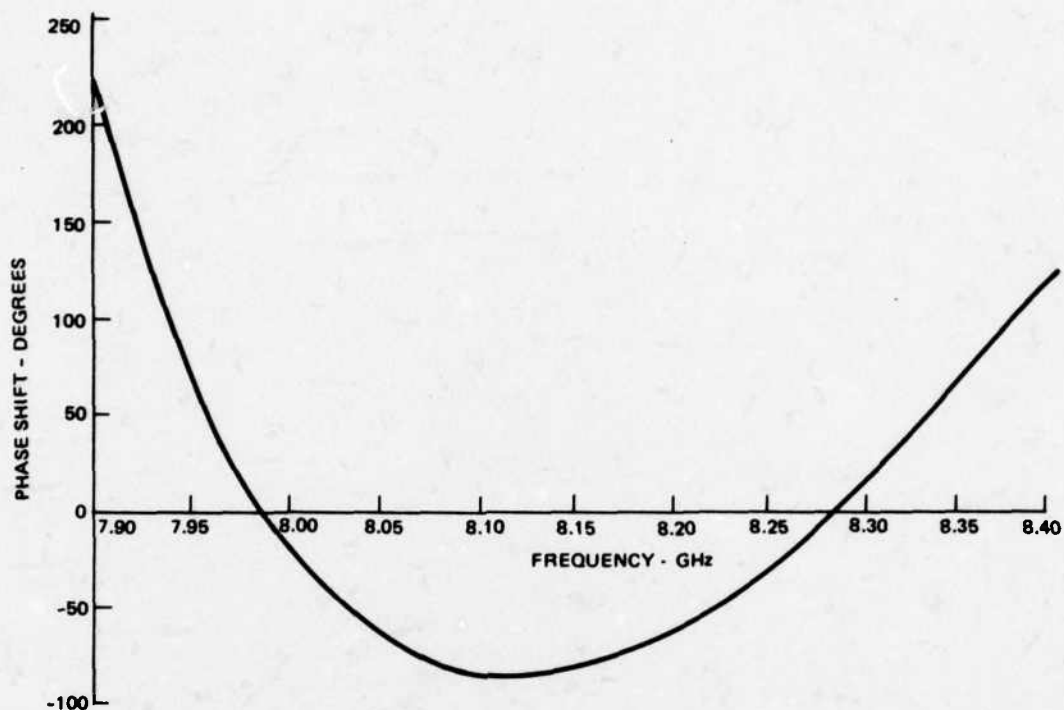
776-3839
UNCLASSIFIED

Figure 2-10. AN/FSC-78 Transmit High Pass Filter

At 700 MHz, the transmitter or receiver is specified to have a phase linearity of ±0.15 radian over 60 MHz bandpass and ±0.4 radian over 125 MHz. The amplitude response is ±1 dB over 60 MHz bandpass and ±2 dB over 125 MHz.

2.6   AN/TSC-86 CONTINGENCY TERMINAL

The information available on the Communication Terminal, Satellite AN/TSC-86 is from the Design Plan. The phase distortion budget is summarized in Table 2-I, assuming equalization in the down converter and the up converter. Two bandwidths, 30 MHz and 40 MHz, are assumed.

Table 2-I. Phase Distortion Budget of AN/TSC-86

	Total Parabolic		Total Cubic		Ripple		Total Phase Distortion	
	30 MHz	40 MHz	30 MHz	40 MHz	30 MHz	40 MHz	30 MHz	40 MHz
Receiver	$\pm2.5^{\circ}$	$\pm5^{\circ}$	$\pm0.9^{\circ}$	$\pm2^{\circ}$	$\pm2.2^{\circ}$	$\pm3.9^{\circ}$	$\pm5.6^{\circ}$	$\pm10.9^{\circ}$
Transmitter	$\pm2^{\circ}$	$\pm4^{\circ}$	$\pm1.7^{\circ}$	$\pm4^{\circ}$	$\pm2^{\circ}$	$\pm3.4^{\circ}$	$\pm5.7^{\circ}$	$\pm11.4^{\circ}$
Specification							$\pm5.7^{\circ}$	$\pm14.3^{\circ}$

2.7   DSCS PHASE II SATELLITE

Phase distortion in the DSCS Phase II satellite is measured by integrating group delay for several gain settings and input power levels, with no significant differences noted for any of these conditions.

The measured phase distortion for channel 2, narrow coverage receive-earth coverage transmit, is shown in Figure 2-11, indicating a ripple of about $\pm 5°$. The measured phase distortion for channel 1, earth coverage receive - earth coverage transmit, is shown in Figure 2-12. There is a ripple of a few degrees in midband, with greater phase distortion at the band edges. Figure 2-13 plots the phase distortion for the worst 40 MHz bandwidth. The distortion is predominately parabolic, reaching about $12°$ at an offset of 15 MHz from center.

776-3840
UNCLASSIFIED
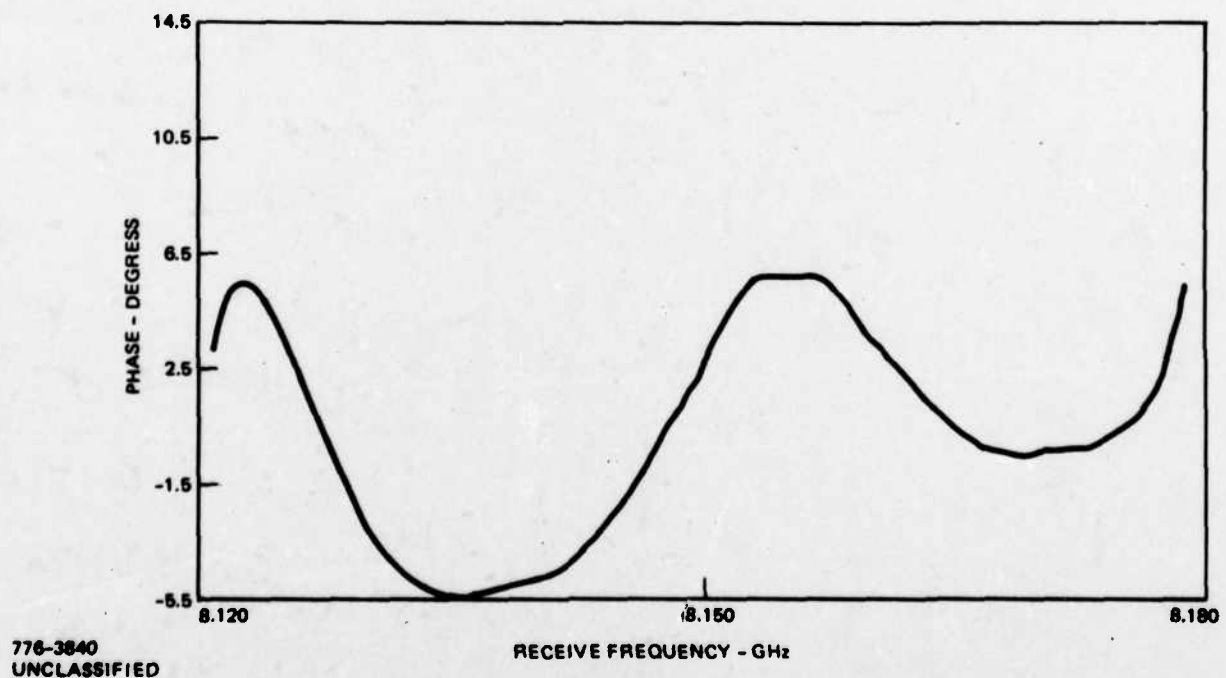
Figure 2-11. DSCS-II Phase Distortion, NC → EC, 8.125 GHz to 8.175 GHz
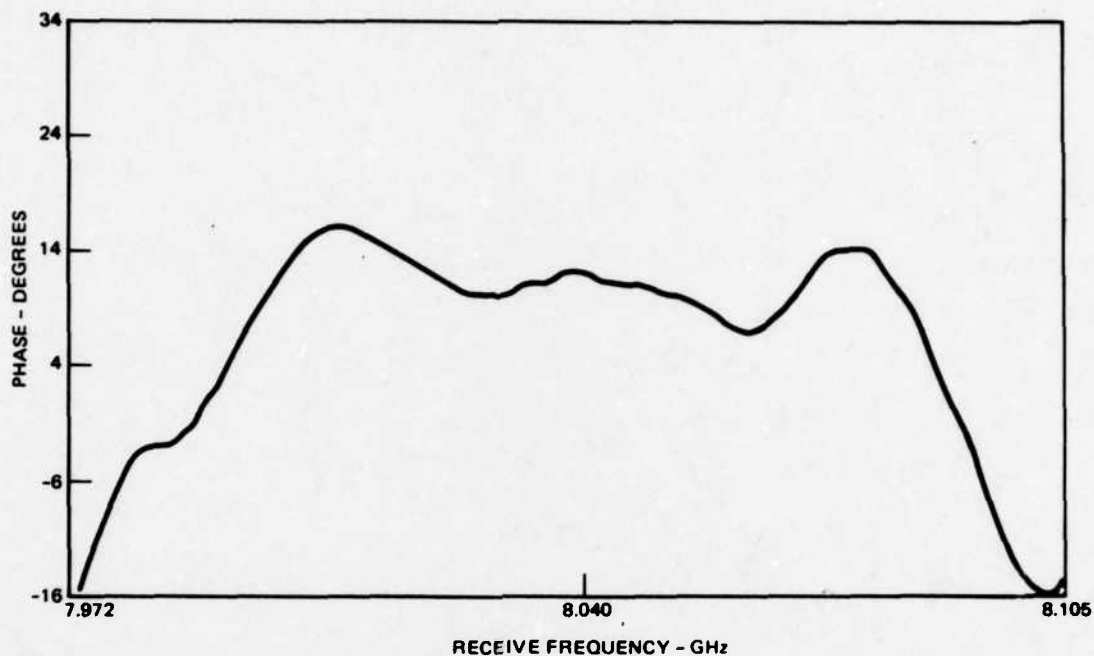
Figure 2-12. DSCS-II Phase Distortion, EC → EC, 7.975 GHz to 8.105 GHz
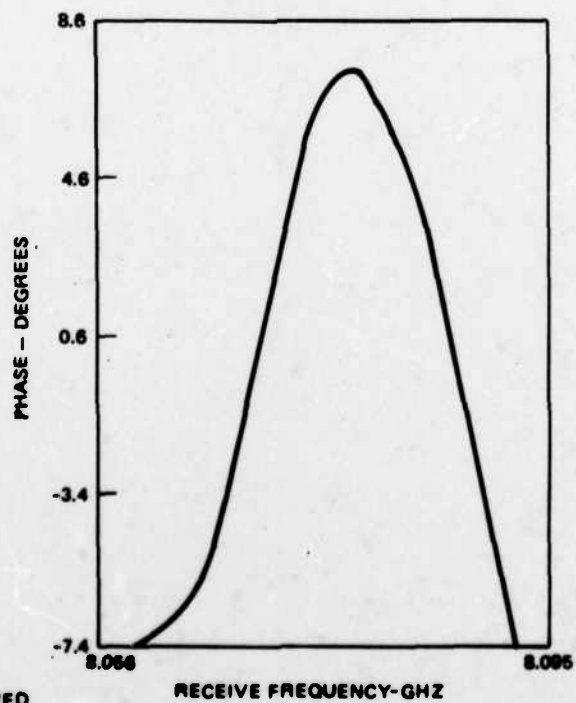
Figure 2-13. DSCS-II Phase Distortion EC → EC, 7.975 GHz to 8.100 GHz
Worst 40 MHz Band

2.8   DSCS PHASE III SATELLITE

The DSCS Phase III satellite has six channels; four of 60 MHz bandwidth, one of 50 MHz bandwidth, and one of 70 MHz bandwidth. Deviation from phase linearity is specified not to exceed $\pm 7.5^{\circ}$ within the central 40 MHz of all channels. In the 70 MHz bandwidth channel, this phase linearity specification applied within any 40 MHz band of the central 50 MHz.

SECTION III
CHANNEL MODELLING

Computer program SIMA was created to simulate digital communication channels permitting the evaluation of system performance and parametric tradeoffs for various system configurations. The program has been written in Fortran IV with a structure having numerous subprogram modules performing various system element functions. In addition, effort has been devoted to making the program as flexible as possible so that the user can easily change parameter values or reconfigure the system by means of alphanumeric inputs while running SIMA.

The overall structure of the system modelled by SIMA is given in Figure 3-1. Any combination of these system elements can be used in a given simulation run with element and system parameters designated by the user through namelist input statements. A full description of the system software, program parameters, and operational considerations is given in Appendix A.

Some features of the program are:

- Single or multiple carrier operation.
- Biphase (BPSK), quadriphase (QPSK) or staggered quadriphase (SQPSK) operation.

- Convolutional encoding/Viterbi decoding.
- IF adaptive equalization.
- Integrate-and-dump or sampling detector.
- Nonlinear amplification (TWT, hard limiter, and soft limiter with AM-PM conversion models).

- Multiple transmitter and receiver filtering (Butterworth, Chebyshev, and Gaussian).
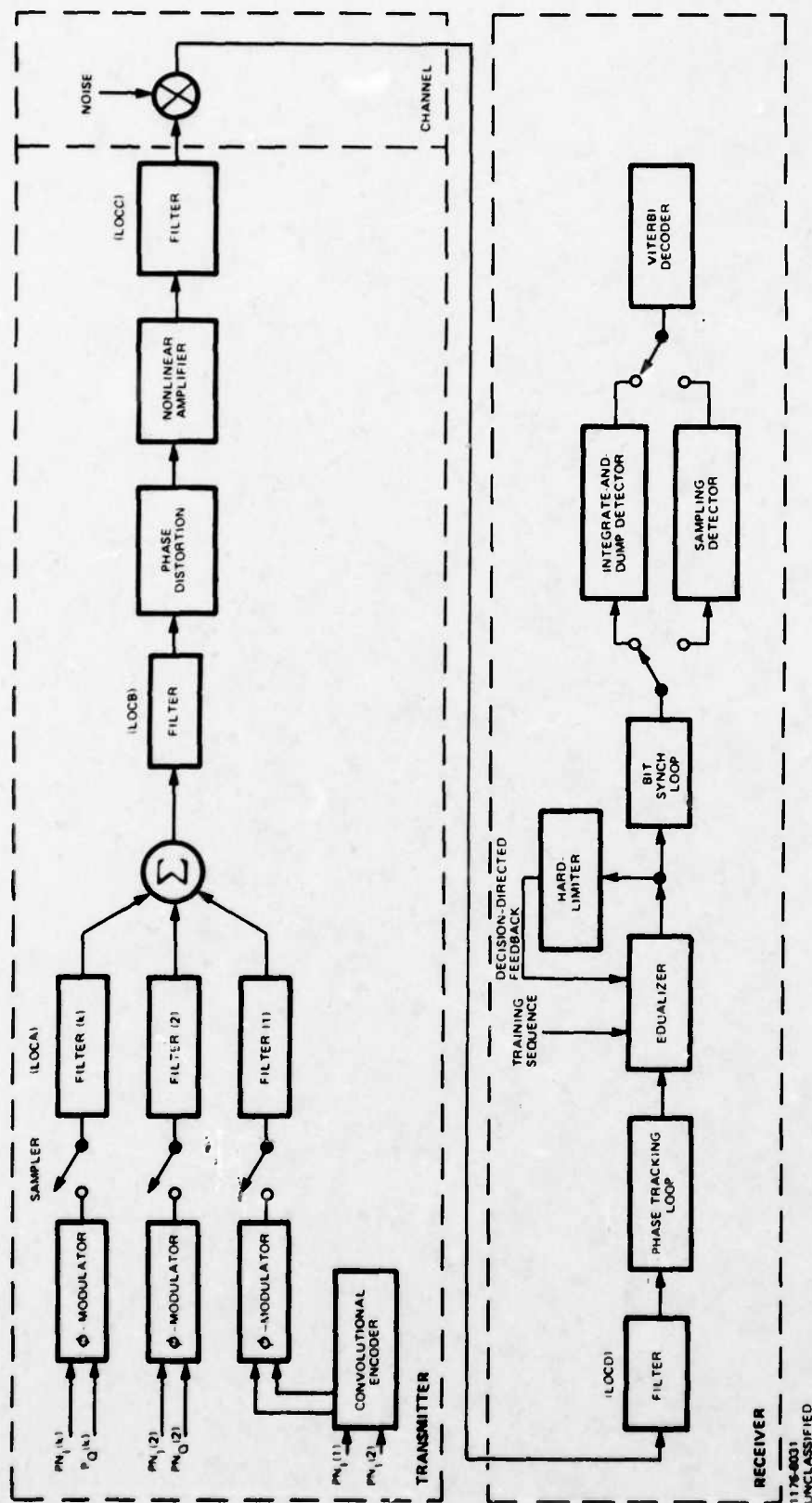
Figure 3-1. Block Diagram of Overall Communication System Modelled

- Channel phase distortion can be introduced.
- Modulator anomalies can be simulated.
- Receiver functions of automatic gain control, phase tracking, and bit synchronization are included but can be defeated.

3.1   SIMULATION TECHNICAL CONSIDERATIONS

The approach used for SIMA involves the generation of sampled-time PN (pseudonoise) sequences. These sampled time sequences are either biphase or quadriphase modulated and subsequently handled as complex baseband data, so that the arbitrary carrier frequency is suppressed in the representation. The data is handled in the time domain for the encoder, modulator, amplifiers, equalizers, detector and decoder and in the frequency domain for system filters and phase distortion elements. Conversion of the data between the time and frequency domains is performed by the Fast Fourier Transform (FFT). This section covers many of the considerations involved in the choice of program parameters for computational accuracy. These considerations include the effects of sequence length, sampling rate, and Fast Fourier Transform, aliasing, and the choice of data sequences. An overall block diagram of the system simulated is given in Figure 3-1.

3.1.1   SEQUENCE LENGTH

Figure 3-2 depicts a binary signal extending from t_o to $t_o + \Delta t$. The duration of this signal is Δt. In the frequency domain this signal is expressed by,

$$F(f) = \int_{t_o}^{t_o + \Delta t} e^{-j\omega t} \, dt = e^{-j\omega t_o - j\omega \Delta t/2} \, \Delta t \, \text{sinc} \, (\Delta \omega t/2), \qquad (1)$$

where $\text{sinc} \, x = \dfrac{\sin x}{x}$.

Similarly a binary message consisting of N bits becomes,

$$F(f) = \Delta t \, \text{sinc} \, (\omega \Delta t/2) \, e^{-j\omega t_o} \sum_{i=1}^{N} (-1)^{k_i} e^{-j\omega(i-1/2)\Delta t} \qquad (2)$$

where $(-1)^{k_i}$ is the polarity of the ith bit with
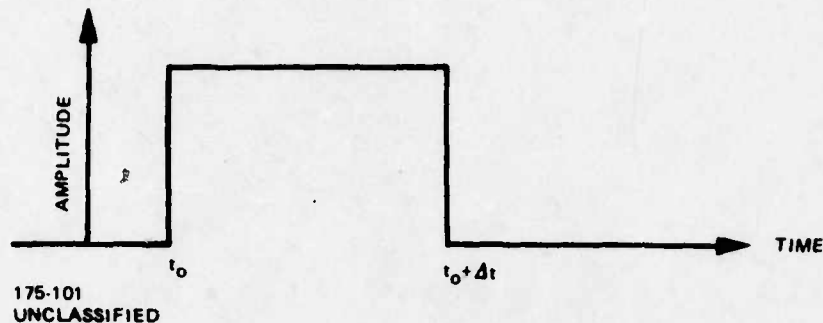
Figure 3-2.   Time Representation of a Binary Signal Bit

$$k_i = 0 \text{ for positive bits}$$

$$= 1 \text{ for negative bits}$$

(3)

and the first signal is assumed to originate at t_o.

The absolute value of the function, sinc $(\omega\Delta t/2)$, is represented by figure 3-3. For convenience of subsequent references $|\text{sinc}(x)|$ is also shown in the same figure. The nth null of this function occurs at $x = \pm n\pi$. Since $x = 2\pi f\Delta t/2$, the nth null also occurs at $f = n/\Delta t$.
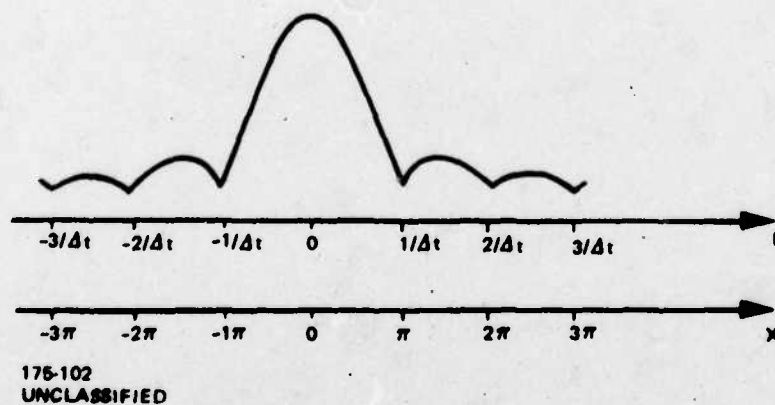
Figure 3-3. $|\text{Sinc }(\omega\Delta t/2)|$ and $|\text{sinc }(x)|$ Functions

For the sake of simplicity, but without loss of generality, henceforth t_o appearing in (1) and (2) is set to zero. Observe from (1) and (2) that the frequency spectrum is continuous. An infinite number of sample points are needed to describe a nonperiodic infinite length signal. Such signals are not well suited for computational analysis.

From the theory of Fourier Transforms, when a periodic waveform (with period T_b) is represented by a baseband signal, the spectral density of this signal can be nonzero only at $f = \pm n/T_b$; $n = 0, 1, 2, \ldots,$ as shown in Figure 3-4. A periodic waveform is best suited for analysis by a digital computer.

3.1.2   INPUT DATA SEQUENCES

SIMA approximates an infinite length digital sequence by repeating periodic sequences. The number of bits that constitute the periodic sequence is called the "sequence length". The time waveform is oversampled and the time between successive samples is called the "sampling time".

The computer simulation requires data patterns of sufficient length to allow for the occurrence of all significant intersymbol interference patterns with the relative frequency expected for independent bit streams. A maximal length shift register ("pseudonoise" or "PN") sequence has this property. Using a PN sequence of length $2^N - 1$ bits, the sequence taken periodically contains all $2^N - 1$ possible bit patterns of length N in equal frequency. If independent adjacent quadriphase channel sequences are required, a different PN sequence can be used for that channel or the same sequence can be used in a shifted version.

These sequences optimally account for intersymbol interference within their prescribed length constraint and larger sequence lengths more accurately predict intersymbol interference effects as filter bandwidths become narrow compared to the bit rate. Use of the PN sequence and the discrete Fourier transform in the simulation eliminates bothersome "end effects" since all computations are performed modulo the sequence length.

SIMA has provisions for delaying the Q (quadriphase) channel PN sequence with respect to the I (in-phase) channel sequence. With the delay, intersymbol interference patterns generated at the end of the block loop back to the beginning so that all data bit patterns in the Q channel still have equal weight. Staggered quadriphase (SQPSK) can also be simulated by delaying the Q data one half bit time from the I data.
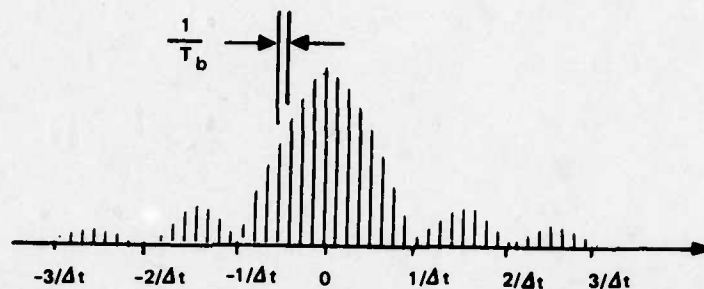
Figure 3-4. Nonzero Frequency Values are Separated By
The Inverse of Time Period, T_b
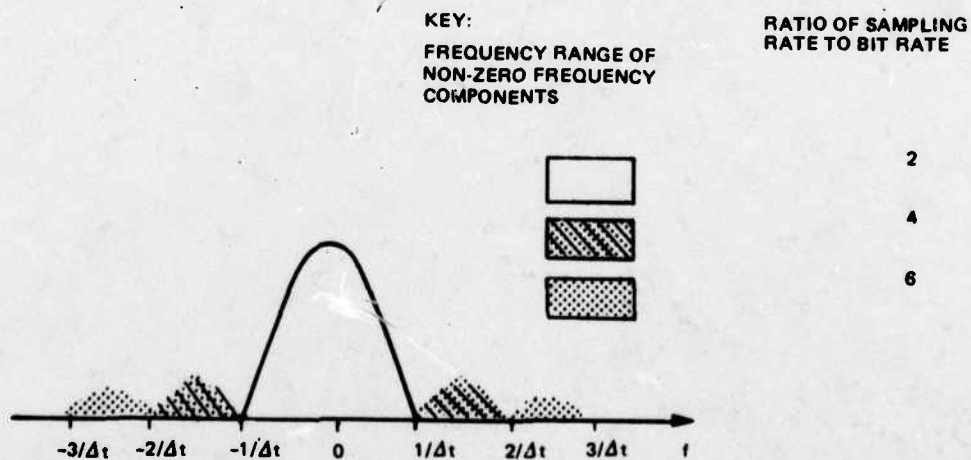
Figure 3-5. Effect of Sampling Rate on Signal Distortion

The selected PN sequences are sampled at equidistant points with the total number of samples = NSMP, the sampling value specified by the user. With a 32 length sequence and 1024 samples, each bit is sampled 32 times. The sampling time interval τ is determined by the bit rate BR and, in the above case is

$$\tau = 32/(1024 \times BR)$$

The user specifies the bit rate BR (or as an alternative the bit time TBD = 1/BR) and all system bandwidths must be entered relative to this bit rate.

The above assumes a constant sampling rate with the number of samples per sequence increasing proportionally to the increase in sequence length. This increase obviously entails a corresponding increase in computer execution time and storage.

3.1.3 SAMPLING RATE

The fundamental theorem of sampling theory states, "if a time function F (t) contains no frequency components higher than F cps, then the time function can be completely determined by specifying its ordinates at a series of points spaced every 1/2F seconds or less." When nonzero frequency components occupy the entire spectrum from 0 to F, then the minimum required sampling rate to prevent distortion is 1/2F. It follows then that sampling a baseband signal at a rate 1/2F will neglect all frequency components above F.

In the present context sampling rate refers to a numerical evaluation process. To transform the signal from time to frequency domain, the computer takes s time samples per bit; s being the sampling rate normalized with respect to bit rate. Figure 3-5 illustrates the "filtering" effect of finite sampling rate. Sampling at twice the bit rate (s=2) has the effect of "ideal" first null filtering. All frequency components outside the darkened area are set to zero due to insufficient sampling rate. As the sampling rate increases, the resulting spectrum more closely approximates the actual unfiltered spectrum. Sampling at 6 times the bit rate eliminates the spectra only past the third null.

The required sampling rate (relative to the bit rate) is determined by the kind of filtering performed in the transmission link, and by the non-linear amplifiers following these filters. Non-linear amplifiers usually generate intermodulation products; hence they "spread" the signal spectrum in frequency domain. This spreading, not the computational errors, determines the spectral occupancy of the signal.

The numerical sampling used to perform the conventional Fourier Transform has the effect of an ideal bandpass filter. The "filter" cut-off frequency is at $\pm(s\Delta t/2)$. s (not necessarily an integer) is the average number of computational time samples per bit duration Δt. Throughout, a uniform sampling rate is assumed. Since the "filter" is ideal, no signal distortion occurs at frequencies below cut-off. Actually, the present program employs Fast Fourier Transformation (FFT). It is shown below that the FFT does have the same cut-off characteristics as the conventional FT, but also introduces approximation errors at frequencies below cut-off.

3.1.4   FAST FOURIER TRANSFORMATION (FFT)

As implied by its name, the FFT saves significant computer time over conventional Fourier Transformation. This saving does not come freely. Above, it was shown that insufficient sampling rate results in non-ideal filtering when conventional Fourier Transformation is used for signal conversions between time and frequency domains. It will be shown that the FFT introduces additional errors due to "aliasing".

The mathematical treatment of the discrete Fourier transform follows, where Δt is the sampling time spacing, Δf is the sampling frequency spacing, N is the number of sample points used, and $x(m\Delta t)$ is the mth data sample.

$$x(m\Delta t) = \sum_{n=\frac{-N}{2}+1}^{N/2} X(n\Delta\omega)\,\Delta f e^{jnm\Delta\omega\Delta t} \tag{4}$$

choose $\Delta\omega$ and Δt so that $\Delta\omega\Delta t = 2\pi/N$:

$$x(m\Delta t) = \sum_{\frac{-N}{2}+1}^{N/2} X(n\Delta\omega)\,\Delta f e^{jnm\frac{2\pi}{N}} \tag{5}$$

$$= \sum_{0}^{N/2} X(n\Delta\omega)\,\Delta f e^{jnm\,2\pi/N}$$

$$+ \sum_{\frac{N}{2}+1}^{N-1} X((n-N)\Delta\omega)\,\Delta f e^{j(n-N)m\frac{2\pi}{N}} \tag{6}$$

Then $-Nm \frac{2\pi}{N}$ can be dropped in the second exponent by periodicity. Let

$$X_m = x(m\Delta t) \tag{7}$$

and use the following notation for the transformed quantities:

$$X_n = \begin{cases} X(n\Delta\omega) \Delta f & , \quad 0 \leq n \leq \frac{N}{2} \\ X((n-N)\Delta\omega) \Delta f & , \quad \frac{N}{2}+1 \leq n \leq N-1 \end{cases} \tag{8}$$

etc. This is equivalent to taking the negative frequency portion of the transformed values and shifting it upward by $N\Delta\omega$. Then:

$$X_m = \sum_{n=0}^{N-1} X_n e^{jnm \frac{2\pi}{N}} \tag{9}$$

This is just the discrete Fourier (inverse) transform and can be evaluated rapidly by the Fast Fourier transform algorithm. Including the forward and inverse transforms, computations proportional to NlogN are required. For N samples in the time domain, computations proportional to N^2 are required. Except for very simple filter structures, the constant of proportionality to N is greater for the all-time domain method than the constant of proportionality times log N for the Fourier transform method. (Note that it is desirable to make the block size N as small as possible for a given number of bits, M, to reduce the log N factor.) For a given data bit sequence of length M, there are N/M samples per bit.

The discrete Fourier transform is periodic in the sense that

$$X_{m+N} = X_m \tag{10}$$

Although one is really only interested in X_m for m = 0, 1, ..., N-1, the importance of this property is that end effects are eliminated and intersymbol interference patterns generated at the end of the block loop back to the beginning. The net effect (for N reasonably large compared to the filter time constant) is that of a system where the same data sequence is sent over and over again.

The FFT produces a repetitive frequency spectra. The frequency separation between these spectra is equal to the number of samples per bit divided by the bit time. As shown by Figure 3-3, the nth null of the sinc $(w\Delta t/2)$ spectrum occurs at $f = n/\Delta t$; Δt being the bit duration. (Throughout, baseband signals are assumed.)

3-9

Hence, the number of computational time samples per second determines the aliasing frequency. Equivalently, the number of samples per bit duration, s, determines the aliasing frequency normalized with respect to the first null of the frequency spectrum.

The "aliasing" phenomenon is illustrated in figure 3-6. Figure 3-6A shows a finite bandwidth spectrum with cut-off frequency, $f_{c.o.} \simeq \pm 2.7/\Delta t$. When the time domain equivalent of this spectrum is converted to the frequency domain using conventional Fourier Transform on a digital computer, figure 3-6B results. Figure 3-6B assumes that the computer calculates the frequency response by sampling the time function at a uniform rate, $s = 3$. Figure 3-6C displays the resulting spectrum when the FFT computer routine is used instead of the regular Fourier transform routine. Because FFT introduces repetitive frequency spectra, the tails of the adjacent (erroneous) spectra fall into the actual signal spectrum introducing computational errors. As shown by figure 3-6D, the FFT truncates the frequency spectrum to a signal period of the repetitive sequence, but only after introducing aliasing errors due to spillover from physically nonexistent spectra; that is, from spectra that only exist for computational ease.

Parameter s gives the total frequency range covered in the simulation if bit rate is unity. As an example, if NSMP = 512 and LSEQ = 31, $s = \frac{512}{31} \simeq 16$ Hz. Since the bandwidth is two-sided this represents the frequency range [-8 Hz, 8 Hz]. For a non-unity bit rate, BR, the frequency range covered equals $s \cdot BR$. Spectral energy lying outside the range $[\frac{-s \cdot BR}{2}, \frac{s \cdot BR}{2}]$ is aliased back and added to the true value to produce computational errors. For example, energy at frequency $s \cdot BR$ aliases to appear at frequency f=0. In a linear system the aliasing is minimal since the spectral energy of the PN data falls off relatively fast and system filter bandwidths further minimize the out of band response.
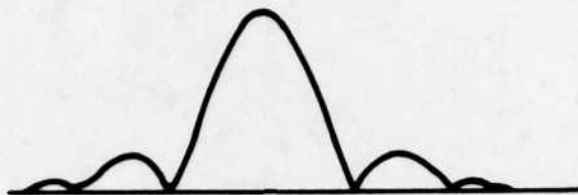
Figure 3-6A. Filtered Signal Spectrum
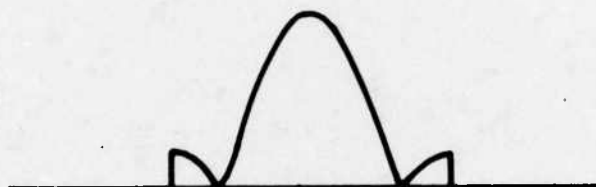
Figure 3-6B. Filtered Signal Spectrum
After Computational Fourier Transform, S=3
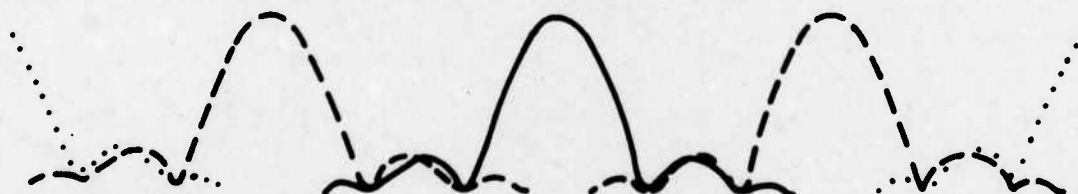
Figure 3-6C. Filtered Signal Spectrum in FFT
Representation, Sampling Rate S=3

COMPUTATIONAL ERROR
CAUSED BY "ALIASING"

Figure 3-6D. Filtered Signal Spectrum After
Computational FFT, S=3

Figure 3-6. Generation of Aliasing Error in FFT

3-11

3.2   CHANNEL ELEMENT MODELS

This section details the methods used for simulating the various system elements used in the channel. Models are given, including the technical considerations involved in the selection of these models and their parameters.

3.2.1   SIGNAL GENERATION AND MODULATION

This section details the manner in which the input data sequences to the in-phase (I) and quadrature (Q) channels of a given signal are generated, sampled, and phase modulated. In addition, the technique for generating multiple accessing signals is discussed. The following parameters and arrays used in SIMA govern the signal generation and format of the signals.

Parameter		Description
NSMP	–	The number of total signal samples used in the simulation
LSEQ	–	The length of the PN sequences generated for both the I and Q data channels
NSB	–	The number of samples per bit used (NSB = NSMP/LSEQ)
QDEL	–	The relative delay between bits in the I and Q channels (given in fraction of a bit)
MPSK	–	The type of phase modulation used (=2) biphase (=4) quadriphase (QPSK) or staggered quadriphase (SQPSK)
ISQ	–	Indicates whether QPSK or SQPSK (=0) QPSK (=1) SQPSK – implies QDEL = .5
G(4)	–	Four values of modulator gain
P(4)	–	Four values of modulator phase locations (nominally 0°, 90°, 180°, 270°)
NCHAN	–	Number of accessing signals in the communications channel
MVIT	–	Determines whether PN data sequences will be convolutionally encoded (and Viterbi decoded) (=0) no encoding (=1) I and Q channels encoded independently (=2) I and Q channels encoded in an interleaved manner
Z(NSMP)	–	Array of the NSMP complex signal samples
FCENT	–	Specifies center frequency offset (normalized by the bit rate) of the accessing signals
CPR(NCHAN)	–	Specifies power in each accessing signal

3.2.2   GENERATION OF PN SEQUENCES

Subroutine XCOD is used to generate the PN data sequences required. Function MSRG is a function subprogram which simulates a modular shift register binary code generator. XCOD generates an initial state for the shift register and the feedback tap vector ITAP to generate a maximal length sequence of LSEQ bits and then calls MSRG for the I and Q channels. Care is taken to insure that different PN sequences are used for the I and Q channels to eliminate performance anomalies that could result from correlation effects if the same sequences were used. The PN sequences are returned in arrays RI and RQ.

If encoding is specified by the user, subroutine ENCODE is called to convolutionally encode RI and RQ as required by the encoding parameters. The PN sequences originally in RI and RQ are replaced by the encoded sequences. In either case arrays RI and RQ replace the sequence of 0 and 1 values by the values −1 and +1 for modulation.

3.2.3   SIGNAL SAMPLING

The LSEQ bit values in arrays RI and RQ are time-sampled to NSB samples per bit and then stored in the array Z of dimension NSMP. If shifting parameter QDEL is not zero, the sampling of the Q channel is delayed by QDEL · NSB samples. For SQPSK, the sampling of the Q channel is delayed by NSB/2 samples.

If adaptive equalization with a training sequence is to be performed, the desired response D of dimension NSMP is set up with the I samples in the real part of D and the Q samples in the imaginary part of D.

3.2.4   PHASE MODULATION

The sampled-time data sequences are input to a modulator (subroutine CMOD) which can be user-specified as biphase or quadriphase. For quadriphase, different PN sequences are used in the in-phase and quadrature channels. The modulator outputs are

$$\alpha_1 e^{j\beta 1}$$
$$\alpha_2 e^{j(\frac{\pi}{2} + \beta_2)}$$
$$\alpha_3 e^{j(\pi + \beta_3)}$$
$$\alpha_4 e^{j(\frac{\pi}{2} + \beta_4)}$$

(11)

In the default mode (ideal modulator), $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 1$ and $\beta_1 = \beta_2 = \beta_3 = \beta_4 = 0$. For biphase operation, $\alpha_2 = \alpha_4 = 0$. To simulate the effects of modulator phase and amplitude imbalance, the user can input new values of the α and β parameters through namelist INPT parameters G and P.

Each time-sample of the I and Q data sequences is scaled in amplitude and shifted in phase, as required, to permit the evaluation of channel crosstalk and performance degradation as a function of modulator anomalies. In a communications system, the data modulates an RF carrier. In SIMA, the carrier is assumed to be at a frequency very much larger than the bit rate. Therefore, the center channel data samples are handled as complex quantities at baseband in the simulation. If several carriers are simultaneously accessing the channel, modulation is treated as described in Section 3.2.5.

3.2.5   MULTIPLE ACCESSING CARRIERS

The previous sections have assumed the case of a single phase-modulated carrier passing through the communications system. A typical channel may have multiple signals passing through the channel producing various effects such as AM-PM conversion, signal suppression, and adjacent channel interference. SIMA allows such multiple carrier operation. Most system elements presented in this report operate in the same manner regardless of the number of signals present. The main exception involves changes in the signal generation/modulation section of the program.

With multiple carrier operation, the user inputs the number of carriers desired, NCHAN. The desired composite signal structure is set up subject to the limitation that all carriers have the same amplitude and phase modulation (BPSK, QPSK, or SQPSK). A loop is formed within SIMA which sequentially performs the following operations for each carrier.

a.   PN sequences are generated for the I and Q components of that carrier. Care is taken to insure that all PN sequences generated for the various carriers are different.

b.   These sequences are appropriately sampled and phase modulated in the time domain to form an array of NSMP complex time samples of that carrier at baseband. The bit transition times of the sequences are offset by one-eighth of a bit duration to randomize the composite waveform.

c.   This array of time samples is converted to the frequency domain via the FFT and filtered (if desired) by a filter structure input by the user.

d.   These (filtered) frequency samples are then shifted in the frequency domain to form a signal offset in frequency from baseband by an amount FCENT specified by the user. FCENT is the center frequency of that signal normalized by the bit rate.

e.   The (shifted) frequency samples are converted to the time domain by the FFT, scaled by $\sqrt{CPR}$ specified for that channel, and added to the NSMP time samples already generated for the other carriers.

Operations (a) through (e) are performed for each of the NCHAN carriers so that the final array of time samples contains the composite waveform resulting from the multiple carriers desired. Note that the channel on which detection is performed must be specified as baseband; i.e., no frequency offset. If convolutional encoding is desired by the user, it is performed on that channel specified as baseband.

The composite array of time samples are sequentially passed through the various system elements as for the case of a single carrier. Detection and decoding are performed for only the baseband signal. Operational considerations for multiple carrier operation are given in Appendix A.

One limitation concerns the permissible location of signals which are offset in frequency from the center channel. As an example, consider NSMP = 1024 time samples, and NSB = 32 time samples per bit. The total frequency range covered is from -16 to +16 times the bit rate. Spectral energy lying outside this frequency band is aliased to give erroneous results within the specified frequency range. Although the data spectrum falls off rapidly after the first few nulls, if an adjacent channel is displaced further than about 8 nulls (FCENT = +8) from the center channel, aliasing could become a problem. System filters help to alleviate the magnitude of such aliasing in the desired signal spectrum, however, and it is doubtful that placement of

adjacent channels further than 8 nulls from the desired channel would be necessary in most cases of interest.

3.2.6   FILTERS

Filters can be inserted at four separate system locations within the structure of SIMA. These locations are user-specified as LOCA, LOCB, LOCC, and LOCD (see Figure 3-1). Any combination of these four locations can be input in a given run and the various filters can be of different types with differing parameters.

Filtering is performed in the frequency domain. The sequence of signal time samples Z is transformed to frequency domain samples via the Fast Fourier Transform (FFT) and multiplied by the corresponding filter frequency response samples. This operation is performed in subroutine FDOM.

Three ideal filter types can be simulated within SIMA; Butterworth, Chebyshev, and Gaussian filters. In addition, zero phase shift filters with the amplitude characteristics of a Butterworth or equalized Butterworth (see Section 7.1) are modelled in the frequency domain. The appropriate filter type and appropriate filter parameters are inserted by the operator through namelist FLTR. Subroutine PPIN computes and stores the ideal filter pole locations. These pole locations are computed for an ideal filter of the type specified with bandwidth normalized to 2π radians. Filter poles are used since the program has been written to permit generation and display of the time response of the filters to impulse, step, or ramp inputs. A typical channel transient response due to filtering is shown in Figure 3-7.

While the modulator output data can be easily filtered in the time domain, the effects of other system filters are simulated in the frequency domain to avoid the lengthy computations that would be required to convolve the digital data samples with the filter impulse response. The input data to the filter is converted to the frequency domain through use of the Fast Fourier Transform and these complex frequency samples are multiplied by the corresponding frequency response samples of the filter as obtained by calls to subroutine FFR.

Care must be taken when transforming between the time and frequency domains to prevent aliasing errors due to replication by the FFT. Such errors are negligible in the current simulation since the sampling intervals have been chosen to cover a sufficiently wide bandwidth so that such aliasing terms are very low at band edge. Therefore, virtually the same results are obtained whether the filtering is performed in the frequency or in the time domains.
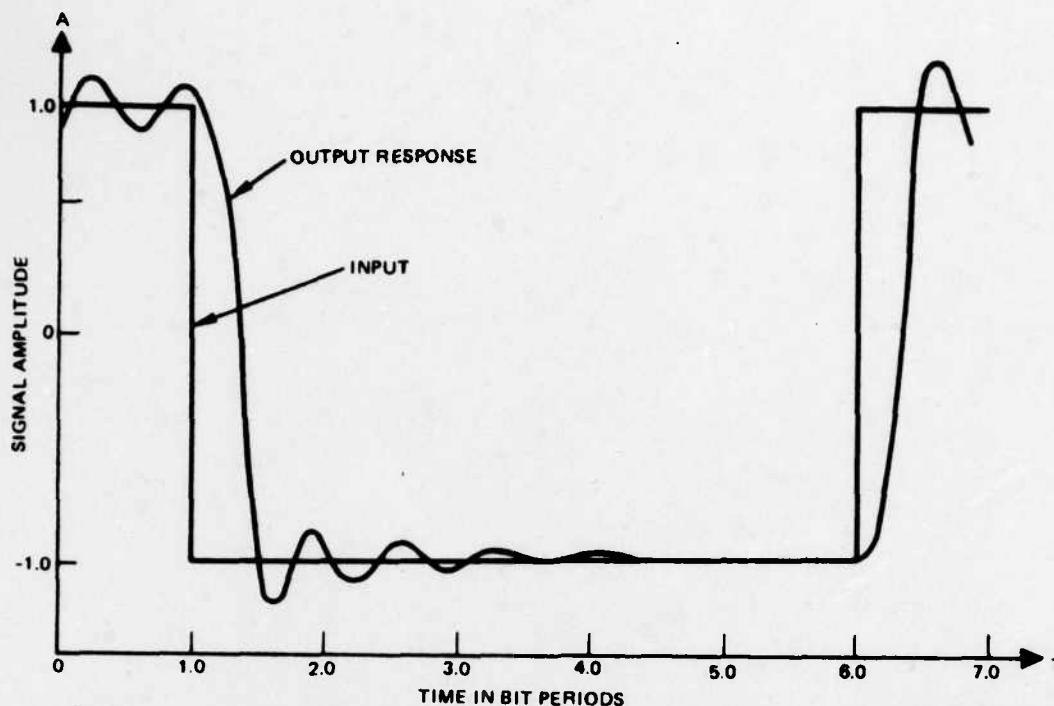
Figure 3-7. Channel Transient Response

3.2.7 EVALUATION OF FILTER POLE LOCATIONS

The pole locations of the various ideal filters are computed as follows (for unity bandwidth):

Butterworth Filters

The analytical form of a Butterworth filter is

$$|F(j\omega)|^2 = 1/(1 + \omega^{2n}) \tag{12}$$

The poles are uniformly distributed on the unit circle in the left half plane and the n pole locations are

$$p_k = \exp\{j(2k-1+n)\pi/2n\}, \quad k=1,2,\ldots,n \tag{13}$$

Chebyshev Filters

The analytical form of a Chebyshev filter is

$$|F(j\omega)|^2 = 1/[1+\delta^2 T_n^2(\omega)] \tag{14}$$

3-17

when δ specifies the ripple amplitude and $T_n(\omega)$ is a Chebyshev polynomial given by

$$T_n(\omega) = \cos\,(n\,\cos^{-1}\omega)\,\text{ for } |\omega| \le 1 \tag{15}$$

Designating the n poles by $P_k = \sigma_k + j\omega_k$, we get

$$\sigma_k = \sinh\,(\tfrac{1}{n}\sinh^{-1}\tfrac{1}{\delta})\,\sin\frac{2k-1}{n}\,\frac{\pi}{2} \tag{16}$$

$$\omega_k = \cosh\,(\tfrac{1}{n}\sinh^{-1}\tfrac{1}{\delta})\,\cos\frac{2k-1}{n}\,\frac{\pi}{2} \tag{17}$$

for $k = 1, 2, \ldots, 2n$, taking only those poles in the left half plane. Another interpretation of these expressions can be obtained by squaring both sides, adding, and dividing by the hyperbolic function to yield

$$\frac{\sigma_k^2}{\sinh^2\,(\tfrac{1}{n}\sinh^{-1}\tfrac{1}{\delta})} + \frac{\omega_k^2}{\cosh^2\,(\tfrac{1}{n}\sinh^{-1}\tfrac{1}{\delta})} = 1 \tag{18}$$

This is the equation of an ellipse in the s-plane with the major axis lying along the $j\omega$ axis.

Gaussian Filter

The Gaussian power density spectrum is

$$|F(j\omega)|^2 = \exp\,\{-\tfrac{1}{2}\,(2\,\ell n\,2)\,\omega^2\} \tag{19}$$

In the frequency domain, equation (19) could be used directly.*

No convenient expression exists for analytically determining the poles of a Gaussian filter so a 3-pole low-pass filter was numerically optimized to synthesize a Gaussian response. Starting with a 3-pole Bessel filter of unity bandwidth, a conjugate-gradient optimization program was used to vary the pole locations so as to most closely approximate a Gaussian filter over the range of $\omega\epsilon$ [-3, 3]. The resulting 3-pole filter has poles at

$$p_1 = -a, \quad p_2 = -b + j\omega_1, \quad p_3 = b - j\omega_1$$

where $a = 1.02437$, $b = .956519$, and $\omega_1 = 1.17703$.

---

*For compatability with generation of the other filter responses and to facilitate generation of step and ramp responses, the filter is computed from stored pole locations.

The peak error deviation between this filter and an ideal Gaussian filter is less than $\pm.7\%$ over all frequencies.

Phase-Distortionless Filters

A five-pole filter having a Butterworth amplitude response with zero phase shift simulates a transversal filter response such as that obtained with surface wave devices at IF. This filter is simulated in the frequency domain by first converting the time samples Z to the frequency domain and multiplying the frequency samples by the corresponding filter response in subroutine FDOM. The response of this filter is given in the frequency domain as

$$\left|F(j\omega)\right|^2 = 1/[1 + (\omega/\omega_c)^{10}]$$

$$\underline{/\!\!\!\!\!\!\!/}\, F(j\omega) = 0 \tag{20}$$

Similarly, the response of the zero-phase equalized filter, described in Section 7.1, is given by

$$G(\omega) = .5\, H(\omega) - 10\,\log_{10}\,[1 + (\omega T_b/\pi)^{10}] \tag{21}$$

where

$$H(\omega) = 20\,\log_{10}\left[\frac{1.115 - .378\cos(\omega T_b) + .128\cos(2\omega T_b) - .046\cos(3\omega T_b)}{.819}\right] \tag{22}$$

with $G(\omega)$ and $H(\omega)$ expressed in decibels.

3.3   NONLINEAR AMPLIFIER

Subroutine AMP models several user-selected nonlinear amplifiers in the time domain. Soft limiters, hard limiters, and two travelling wave tube (TWT) amplitude characteristics are modelled as are four different amplifier phase characteristics. Input signal time samples are phase shifted and scaled in magnitude in accordance with the amplitude and phase types selected and amplifier parameters input by the user. The various amplifier models are described in this section. Any combination of phase and amplitude types is permitted.

3.3.1 AMPLIFIER AMPLITUDE CHARACTERISTICS

3.3.1.1 Hard Limiter

Input TYPE = 3 models the ideal hard limiter producing the saturated amplifier output independent of gain. The transfer function of the hard limiter is given by

$$Z_o = V_{sat} \cdot Z_i / |Z_i| \tag{23}$$

where Z_i is the complex input time sample, Z_o is the complex output time sample, and V_{sat} is the saturated output voltage.

3.3.1.2 Soft Limiter

The soft limiter model (TYPE = 2) assumes a piecewise linear amplifier which has linear amplification until the input signal amplitude exceeds a prescribed level above which hard limiting occurs. The soft limiter model is described by the equations

$$Z_o = \begin{cases} V_{sat} \cdot GI \cdot Z_i / \sqrt{.1} \,, & |GI \cdot Z|^2 \le .1 \\ V_{sat} \cdot Z_i / |Z_i| \quad\quad, & |GI \cdot Z|^2 > .1 \end{cases} \tag{24}$$

where GI is the amplifier gain in the linear region.

3.3.1.3 Traveling Wave Tube (TWT) Models

The primary nonlinear device in a satellite is usually a TWT (TYPE = 1, 4, or 5) used for the high level power amplification. This amplifier exhibits both nonlinear amplification and a linear or nonlinear phase shift. Furthermore, military repeaters usually contain a tunnel diode amplifier limiter (TDAL) or other limiter which controls the characteristics of the repeater in the region of saturation.

The TWT amplifier models are a function of the complex input waveform $Z_i(t)$. The power of this input waveform is P_i. At the input of the amplifier any signal power loss resulting from filtering is removed by normalizing the input power (over the NSMP samples) to unity through a call to AGC. The amplifier operating point is then calculated based upon the output backoff (BO, in dB) entered by the user. TWT backoff is defined as

$$BO = P_o / P_{sat} \tag{25}$$

where P_{sat} is the saturated TWT output power at BO = 0 dB and P_o is the backed-off amplifier output power for a normalized amplifier input power of unity.

3.3.1.4   TWT Amplifier Model (TYPE = 1)

The amplifier gain GI is specified indirectly to produce an amplifier output power P_o for $P_i = 1$, following user specification of the backoff value BO. The amplifier gain GI is a nonlinear function of P_i, P_o, and BO. The gain GI is found by a search procedure to minimize the expression

$$|Z_o - V_o| \text{ for } Z_i = 1 \tag{26}$$

where

$$V_o = (GI \cdot V_i) = \begin{cases} 1, & |GI \cdot V_i| \leq C \\ \dfrac{10^Y}{GI \cdot V_i}, & |GI \cdot V_i| > C \end{cases} \tag{27}$$

$$Y = A \cos [\log_{10} (GI \cdot V_i/D)/B] - A \tag{28}$$

Parameters A, B, C, and D are dependent on the assumed TWT amplifier characteristic. The values used for a general TWT characteristic are:

A = 0.3935
B = 0.4753
C = 0.3548
D = 2.317

These parameters give the characteristic shown in Figure 3-8.

Prior to operation of the subroutine AMP, gain GI is determined by a search procedure so that the amplifier is operating at the correct backoff specified by the user. Following selection of the proper gain GI, input time samples are appropriately amplitude scaled by the equation:

$$Z_o = \begin{cases} V_{sat} \cdot GI \cdot Z_i, & |GI \cdot Z_i| \leq C \\ V_{sat} \cdot Z_i \cdot 10^Y/|Z_i|, & |GI \cdot Z_i| > C \end{cases} \tag{29}$$

where Y is given by equation (28) and A, B, C, and D are the constants previously given.

P_R

OUTPUT POWER RELATIVE TO P_SAT – dB

BACKOFF (dB)

175-292
UNCLASSIFIED
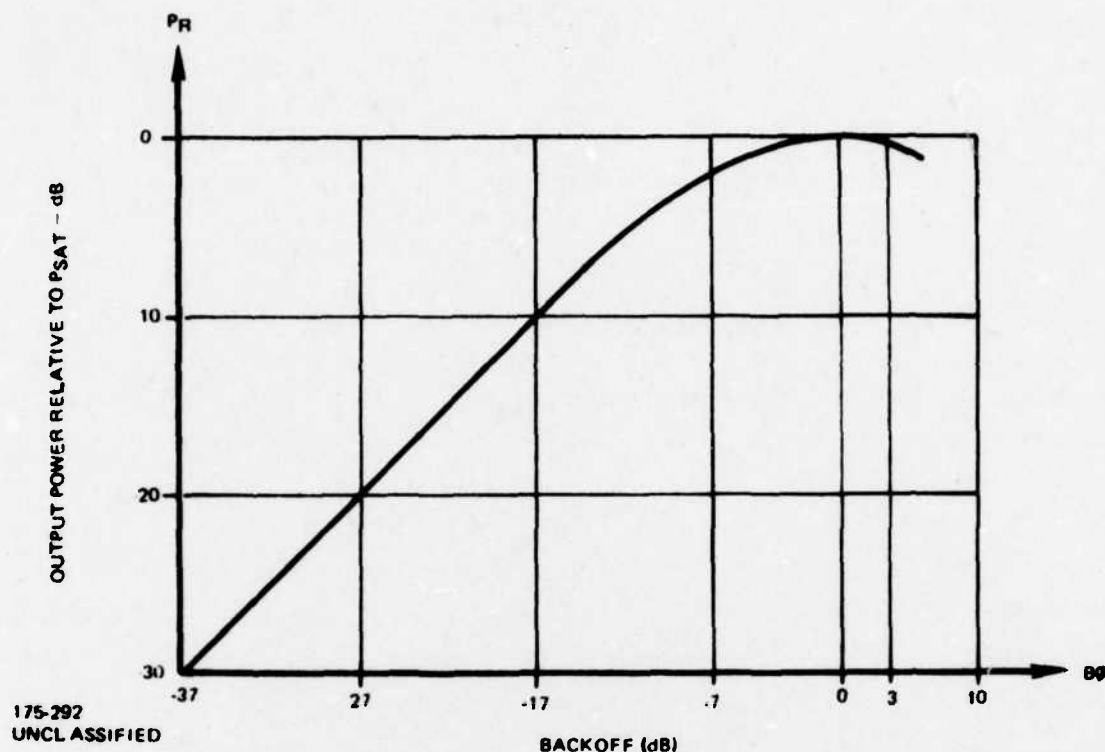
Figure 3-8. Generalized TWT Gain Characteristic Modelled when TYPE = 1

3.3.1.5 Hughes' TWT Model (TYPE = 4)

The amplitude characteristic of the Hughes' TWT for Phase III operation is shown in Figure 3-9 and is modelled in AMP by specification of the input/output values at 5 dB increments over the range of -20 dB < BO < 0 dB. Linear interpolation is used for values between specified points. For BO < -20 dB, linear gain is used. For BO > 0 dB, the amplitude is hard limited to the 0 dB level. Also, the phase shift is held constant for BO > 2 dB.

As noted above, a military repeater will have a limiter, and the gain budget will normally limit the input drive to the TWT such that its maximum output is near the saturation point. Thus, the characteristic is correctly modelled for BO < 0 dB and is a typical representation for BO > 0 dB.
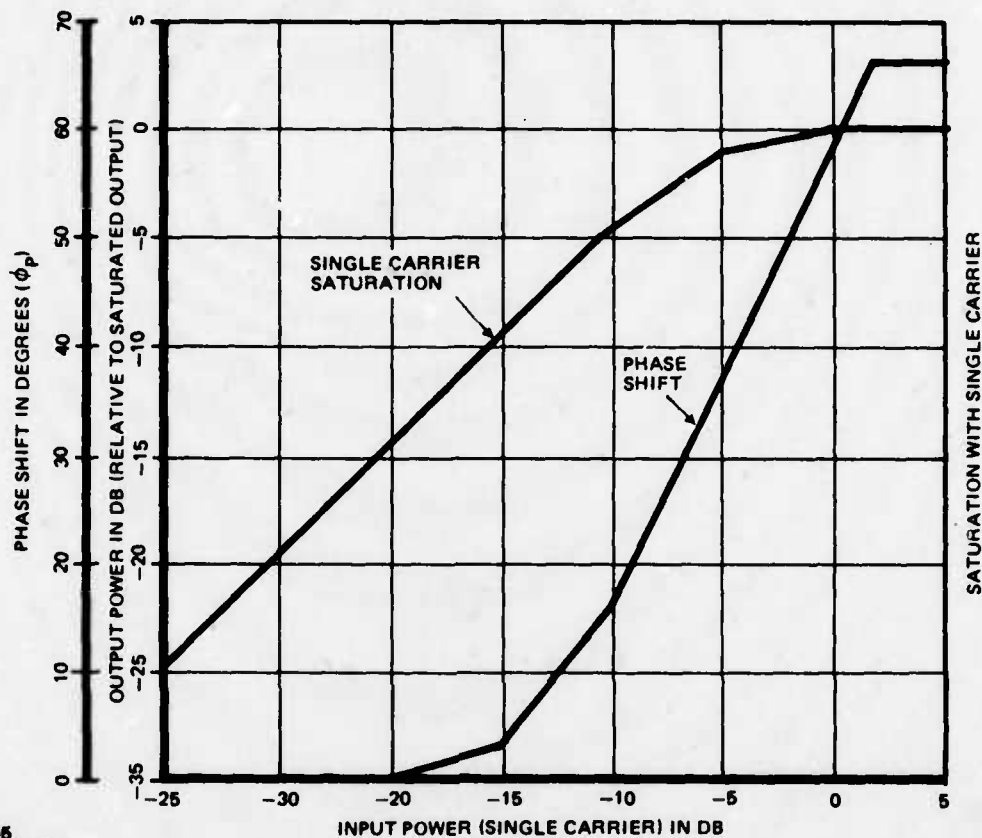
Figure 3-9. Amplitude and Phase Characteristic of the Hughes' TWT

3.3.1.6   Phase II TWT Model (TYPE = 5)

The amplitude characteristic of the Phase II TWT is shown in Figure 3-10 and is modelled in AMP by specification of the input/output values at 5 dB increments over the range of -15 dB < BO < 0 dB. Linear interpolation is used for values between specified points. For BO < -15 dB, a linear gain is used. For BO > 0 dB, the amplitude is hard limited to the 0 dB level.
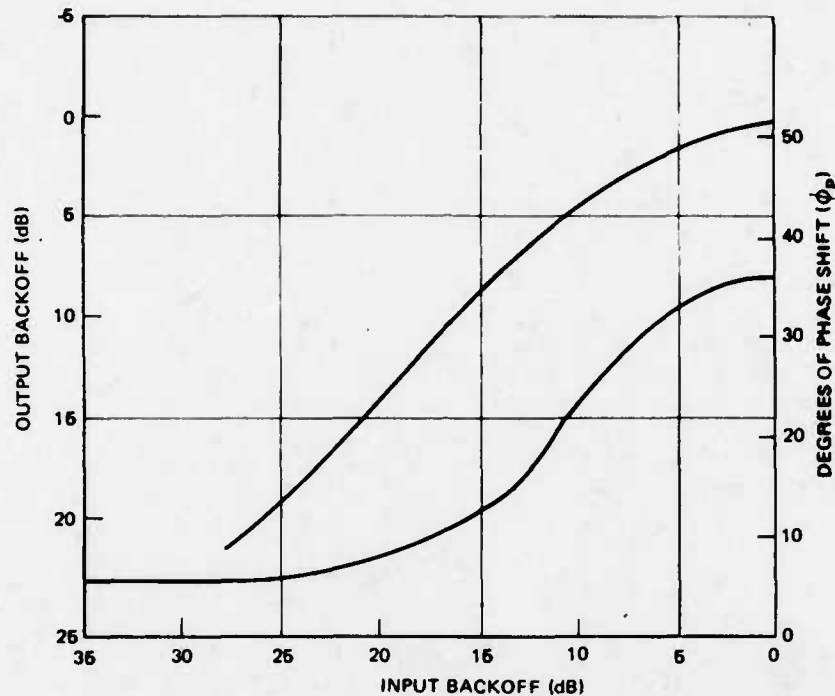
Figure 3-10. Amplitude and Phase Shift Characteristics of Phase II TWT

3.3.2   AMPLIFIER PHASE CHARACTERISTICS

Four amplifier phase characteristics are available. The phase models are independent of the amplitude model selected and phase rotation of each input signal sample follows amplitude scaling of the sample in subroutine AMP. If **PHASE** = 0, no phase rotation occurs. In all other models, the phase shift ω_p is a function of the phase characteristic selected and the amplitude of the input sample. The output sample from subroutine AMP is therefore

$$\hat{Z}_o = Z_o e^{j\omega_p} \tag{30}$$

The AM-PM conversion produced by the amplifier phase shift produces intermodulation effects which degrade system performance.

3-24

3.3.2.1 Berman-Mahle Model (PHASE = 1)

$$\omega_p = a(1 - \exp(-b)|GI \cdot Z_i|^2) + |GI \cdot Z_i|^2/C \tag{31}$$

where a, b, and c have the values

$$
\begin{aligned}
a &= 0.602 \\
b &= 0.660 \\
c &= 102.4
\end{aligned}
$$

3.3.2.2 Linear Phase Model (PHASE = 2)

$$\omega_p = \omega_s \cdot 20 \log_{10}(GI \cdot Z_i) \tag{32}$$

where ω_s is the specified phase shift in radians/dB.

3.3.2.3 Truncated Linear Phase Model (PHASE = 3)

The truncated linear phase is given by the expression

$$\omega_p = \begin{cases} \omega_s \cdot 20 \log_{10}(GI \cdot Z_i) \ , & GI \cdot |Z_i|^2 \geq p_f \\[2ex] 0_o & , & GI \cdot |Z_i|^2 < p_f \end{cases} \tag{33}$$

where p_f is a prescribed input back-off power. At present, $p_f = 0.1$. Some TWT specifications give phase ω_s in degrees/dB while others give max phase, $\omega_{p,\,max}$, at BO = 0 dB. This phase characteristic is given in Figure 3-11.

3.3.2.4 Hughes' TWT Phase Characteristic (PHASE = 4)

The phase shift of the Hughes' TWT as a function of drive is also shown in Figure 3-9 and is modelled in AMP by specification of the phase shift versus input drive in 5 dB increments over the range -20 dB < BO < 0 dB. Linear interpolation is used for values between specified points. For BO < -20 dB, phase shift $\omega_p = 0^o$ and for BO > +2 dB, the phase shift is maintained at 66°.

DEGREES OF PHASE SHIFT (ϕ_P)

INPUT BACKOFF (dB)

P_F
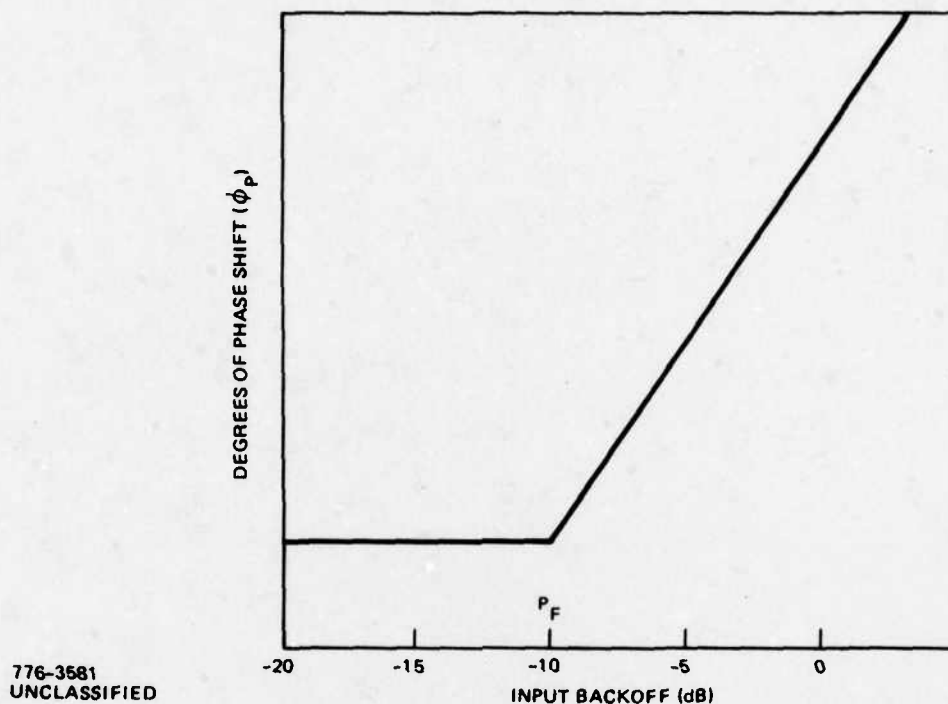
Figure 3-11.   Linear AM-PM Conversion Characteristic (PHASE = 3)

3.3.2.5   Phase II TWT Phase Characteristic (PHASE = 5)

The phase shift of the Phase II TWT as a function of drive is also shown in Figure 3-10 and is modelled in AMP by specification of the phase shift versus input drive in 5 dB increments over the range $-20 \text{ dB} < \text{BO} < 0 \text{ dB}$.   Linear interpolation is used for values between specified points.   For BO < -20 dB, phase shift $\omega_p = 6^O$.   For BO > 0 dB, the phase shift is held at 36°.

3.4   CHANNEL PHASE DISTORTION

Four types of phase distortion are modelled in SIMA.   These types are sinusoidal, cosinusoidal, parabolic, and step (discontinuous) phase.   These phase types are shown in Figure 3-12.   These phase distortions are located in function DIST which returns the resulting phase θ_D at the calling frequency W.   This phase is used to rotate the complex signal samples in Z which have been transformed to the frequency domain.   Each type of phase distortion is described below.
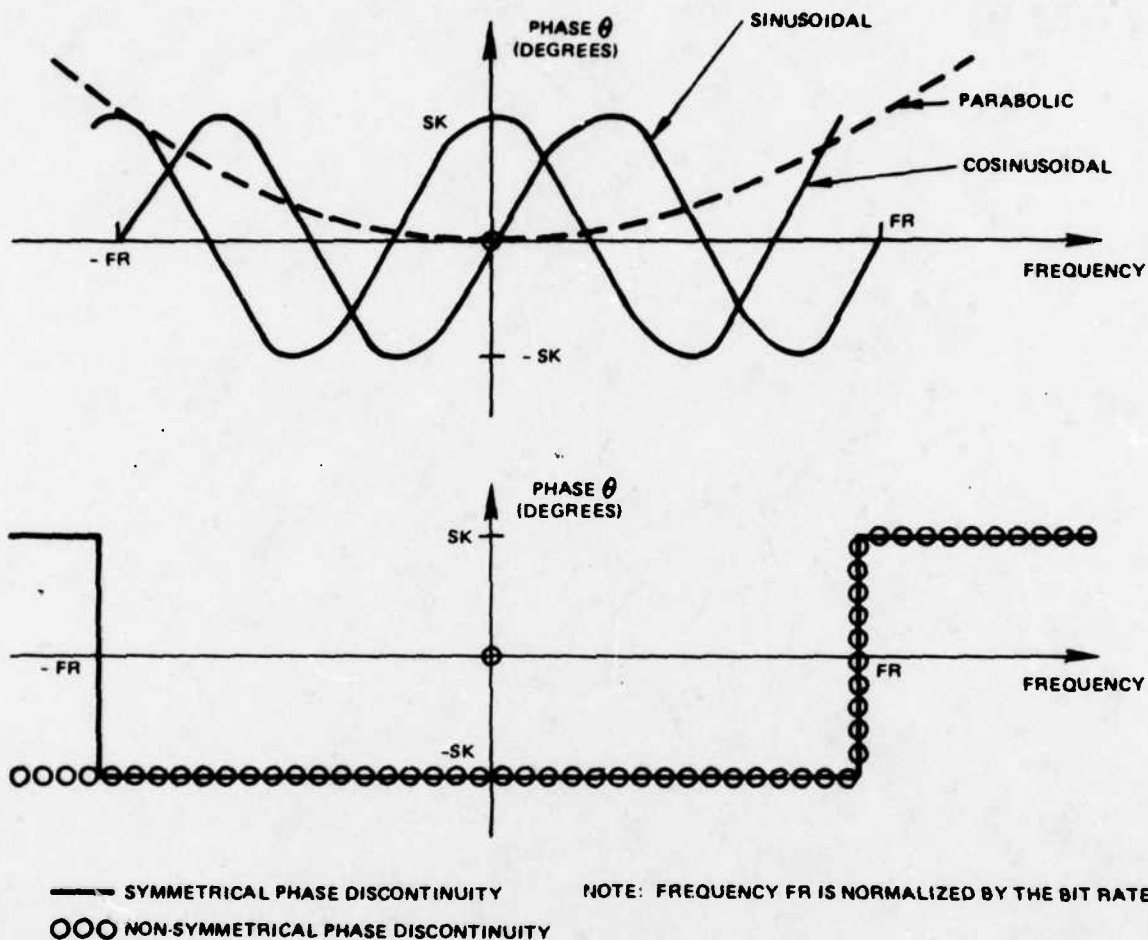
Figure 3-12. Phase Characteristics Modelled

3.4.1   SINUSOIDAL PHASE DISTORTION

The sinusoidal phase distortion of period T_R is given by

$$\theta(\omega) = \beta \sin(\omega T_R) \tag{34}$$

where T_R determines the ripple period in the frequency domain and the phase varies between $\pm\beta$. If T_R equals the bit period TBD, there is one ripple cycle in the first null bandwidth $[0 \leq \omega \leq 2\pi \, \text{TBD}^{-1}]$.

If $s(t)$ is the signal waveform with Fourier transform $S(\omega)$, the time domain response after distortion is

$$r(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} S(\omega) \, e^{j(\omega t + \beta \sin \omega \, T_R)} d\omega \tag{35}$$

$$r(t) = \sum_{k=-\infty}^{\infty} J_k(\beta) \, s(t + k \, T_R) \tag{36}$$

Therefore, paired echos are produced by sinusoidal distortion. For small β, only the $k = \pm 1$ terms contribute significant intersymbol interference. In addition, the echos are antisymmetric. On the average, the slicing level remains at zero so that the noise margin for detecting the bit immediately following a transition is reduced as given by the margin

$$m = J_0(\beta) - 2 J_1(\beta) \cong 1 - \beta \tag{37}$$

while the margin is enhanced for the bit preceding the transition. Because sinusoidal distortion produces a complex symmetric transfer function, BPSK and QPSK are identically degraded and there is no cross-channel interference.

3.4.2   COSINUSOIDAL PHASE DISTORTION

Cosinusoidal phase distortion of period T_R and amplitude β is given by

$$\theta(\omega) = \beta \cos(\omega \, T_R) \tag{38}$$

For small β, the resulting time response is

$$r(t) \cong J_0(\beta) \, s(t) + j \, [J_1(\beta) \, s(t-T_R) + J_1(\beta) \, s(t+T_R)] \tag{39}$$

where the intersymbol interference is in quadrature with the pulse. Therefore, the two QPSK channels interact with the degradation being similar to that produced by sinusoidal distortion. For BPSK, however, the intersymbol interference is in phase quadrature and has no effect on the demodulation unless it affects tracking loops due to the resulting phase steps produced by bit transitions. Note that if T_R = TBD, the bit period, the intersymbol interference falls directly on adjacent pulses.

3.4.3   PARABOLIC PHASE DISTORTION

The assumed parabolic phase characteristic is

$$\theta = \beta \left[\frac{\omega \, T_R}{2\pi} \right]^2 \tag{40}$$

3-28

where β is the phase error developed at (radian) frequency $\omega = 2\pi/T_R$. This phase variation yields the time delay variation

$$\frac{d\theta}{dw} = \beta \frac{T_R^{\cdot 2}}{2\pi^2} \omega \qquad \qquad (41)$$

The response to a data pulse with parabolic distortion is similar to that produced by cosinusoidal distortion since the transfer function is nonsymmetrical. For this reason, the QPSK channels interact to produce degradation greater than that for BPSK.

3.4.4 STEP PHASE DISTORTION

Step phase distortion involves a discontinuity in phase at some frequency (or frequencies) and two types of such phase distortions are used in the simulation:

- Nonsymmetrical: $\phi = \begin{array}{l} \beta, \; f \geq FR \\ -\beta, \; f < FR \end{array}$

- Symmetrical $\phi = \begin{array}{l} \beta, \; |f| \geq FR \\ -\beta, \; |f| < FR \end{array}$ $\qquad$ (42)

where $FR = TBD/T_R$.

As before, the nonsymmetrical step phase distortion produces intersymbol interference effects in phase quadrature producing significant interaction of the QPSK data channels. The symmetrical step phase distortion, on the other hand, produces a combination of inphase-and-quadrature interference effects. The overall effect of such phase distortions could be determined with a Fourier series expansion in which the amplitude of each sinusoidal or cosinusoidal component of the expansion determines the amplitude and location of paired echos.

3.5 AUTOMATIC GAIN CONTROL AND PHASE-TRACKING FUNCTIONS

3.5.1 AUTOMATIC GAIN CONTROL

Prior to the phase loop, an automatic gain control (AGC) loop adjusts the receiver input signal energy to unity to provide the appropriate SNR* level specified. The AGC action is performed by the expression

$$AGC = \sqrt{\left(\sum_{i=1}^{NSMP} Z_i \cdot Z_i^* \right) / NSMP} \qquad (43)$$

---

*SNR is equivalent to E_b/N_o at the receiver input prior to any receiver filtering.

where Z_i^* is the complex conjugate of the ith time sample. Subsequently for BPSK data each time sample is divided by the value AGC, while for QPSK data each time sample is multiplied by the value $(\sqrt{2}/AGC)$.

For multiple accessing signals, the AGC value is computed and then boosted by a factor β to bring the energy of the signal in which probability of error is to be computed to unity. This rescaling of amplitude relates the user-specified received E_b/N_o to the energy in this central signal. If the composite received waveform contains N adjacent signals, with the kth adjacent signal having power P_k and the center signal having power P_c, then

$$\beta = \frac{P_c + \sum_{k=1}^{N} P_k}{P_c} \tag{44}$$

Note that NCHAN = N + 1 and the P_k are given by the values CPR entered by the user for each accessing signal. While the ratio β is strictly accurate only for linear channels, suppression effects in the normal saturating channel are minor and are not accounted for in computing β.

If the receiver contains filters, the effect of the filters on the assumed white Gaussian noise present at the receiver input is computed. Based on a non-saturating receiver with a sampling or integrate-and-dump detector, the decrease in noise energy due to the filters is determined by evaluating and integrating the energy spectral response of the composite receiver filters. The input noise energy is there-after decreased by this factor.

3.5.2   PHASE TRACKING

Program SIMA has provision for simulating a phase-locked loop or a Costas loop with first, second, or third order loop filter. Subroutine PLOOPR is first initialized to set up the loop parameters and to initialize the phase to zero degrees.

In operation, the incoming NSMP complex time samples are applied to the loop input three times. The first two iterations have the function of allowing loop transients to settle while the third iteration is used to generate the phase-locked output samples used for the subsequent synchronization search and detection. This iterative process prevents transient tracking errors from affecting the simulation results.

For biphase signals, the Costas loop tracks on the error

$$e_i = SGN(I) \cdot Q \tag{45}$$

where I is the in-phase sample value and Q is the corresponding quadrature sample value. For quadriphase signals, the Costas loop tracks on the error

$$e_i = SGN(I) \cdot Q - SGN(Q) \cdot I \tag{46}$$

3.6   BIT SYNCHRONIZATION

Following the phase tracking loop, bit synchronization is established by determining the average location of zero crossings in the I channel. The NSMP time samples are passed through the sync loop and the location of zero crossings noted. Since the zero crossings of the original data occur at integer multiples of the number of samples per bit, the zero crossing sample locations (modulo the number of samples per bit) are added and subsequently divided by the number of crossings located to give the synchronization delay used for detection. Note that in the present simulation, the synchronization value obtained is a function of the time delay of the system filters and intersymbol interference effects. Since the present simulation does not add noise samples to the input data samples, averaging over NSMP samples is adequate for obtaining accurate synchronization.

If BPSK or QPSK is used with QDEL = 0, the same value of delay is used for the Q channel as established for the I channel. If staggered quadriphase (SQPSK) is used, the bit sync for the Q channel is established as one half bit delayed from that determined as optimum for the I channel. If QDEL is not zero, the Q channel sync is delayed by the appropriate amount.

The bit sync loop calculations are set to handle the case where zero crossings occur either around zero relative delay or around an integer number of bits. Without care in the modulo operations, an anomalous bit sync could be obtained for these cases. As currently performed, no errors occur in such instances.

3.7   BIT DETECTOR MODELS

Two forms of bit detectors are modelled in SIMA. The first is an integrate-and-dump detector (MDET = 1), and the second is a sampling detector (MDET = 2). Both detector subroutines accept the NSMP time samples in array Z and the previously

computed bit sync value. The LSEQ detected quantities are output to a subroutine CERR which computes the average probability of bit error and system degradation at specified SNR based on these detected bit values.

3.7.1 INTEGRATE-AND-DUMP DETECTOR

SIMA models an ideal integrate-and-dump detector by summing over NSB consecutive samples, beginning at the I sample established by the bit sync loop. The sum over the Q samples begins at the sample used for the I channel plus QDEL · NSB samples. Consecutive groups of NSB samples are summed and the LSEQ sums of I and Q outputs are each divided by NSB and stored in array SUM.

3.7.2 SAMPLING DETECTOR

The sampling detector assumes that the sequence of input time samples will be sampled once every NSB samples and this sequence of LSEQ samples are used to estimate the bits transmitted. The optimum sampling time is a function of system filters and other degradations introduced by the channel.

Subroutine SAMPDT accepts the received input samples Z and uses an optimization algorithm for selecting that sampling time which minimizes the probability of bit error at an input SNR = 10 dB. Once this sampling time is determined, the subroutine samples the Z values once every NSB samples and stores these LSEQ samples of the I and Q channel in array SUM.

Note that the presampling receiver filter is called at location LOCD (see Appendix A) with the desired filter parameters.

3.7.3 PROBABILITY OF BIT ERROR COMPUTATION

The average probability of bit error in the I channel is evaluated in subroutine CERR by taking the average of the complementary Weber error functions computed for the absolute value of each of the I bit sums in array SUM

$$P_E = \frac{1}{LSEQ} \sum_{i=1}^{LSEQ} werfc(\sqrt{2 \cdot SNR} \cdot |SUM(i)|) \tag{47}$$

This process is then performed for the quadrature bit sum values. The degradation from ideal is then given by the expression

$$DEGR(dB) = SNR(dB) - 10 \log_{10} \left\{ (werfc^{-1} (P_e))^2 / 2 \right\} \tag{48}$$

Thus, system degradation is measured by first determining the receiver input signal-to-noise ratio (E_b/N_o) which would produce the same resulting probability of bit error in an ideal system, converting this SNR to decibels, and subtracting this number from the actual receiver input SNR used. The terms SNR and E_b/N_o are equivalent in this report and the noise energy used is based on the AGC normalization of input signal energy to unity (Section 3.5.1) prior to any receiver filtering.

3.8   ADAPTIVE EQUALIZER

Subroutine ADEQ models the adaptive equalizer including the transversal filter, LMS weight adaptation algorithm, and generation of the hard limited decision-directed feedback (DFB) reference signal. Adaptive equalization using the LMS algorithm and methods of decision-directed control are discussed in Section 5. The subroutine accepts as input the NSMP signal sample array X and the NSMP desired signal array D for training sequence operation. Other parameters input to ADEQ are:

NTAP - the number of tap weights in the equalizer.

ISP - the tap spacing given as the number of signal samples per tap.

ISHIFT - the displacement between the input signal sample and the corresponding training sequence sample.

MREF - form of the desired signal; (=0) training sequence; (=1) hard-limited decision-directed feedback reference.

ADEQ forms as output the array X of the NSMP equalized signal samples, the mean square error between the equalized samples and the signal samples input to the modulator, and PWRN which represents the energy loss of the noise in passing through the equalizer and integrate-and-dump detector. The final equalizer tap weights are also displayed for the user.

In operation, the subroutine initializes the tap weights to produce an impulse response so that in the beginning the equalizer output is identical to its input. The transversal filter is modelled so that the successive tap weights multiply input signal samples spaced by ISP samples. These NTAP inputs are multiplied by the NTAP weights to form the instantaneous output quantity S. For training sequence operation, the LMS error quantity E is formed by subtracting the output S from the desired signal sample of D located ISHIFT samples away from the signal sample which is being multiplied by the middle tap weight. For DFB operation, the error E is formed by subtracting S from the quantity $\sqrt{2}\,S/|S|$, where $|S|$ is the magnitude of output S.

For each output S, the error generated iterates the tap weights by the LMS algorithm. Following this update of the weights, the input X is shifted one sample forward through the transversal filter to generate the subsequent output S. The equalizer adapts over a total of 12 · NSMP samples by treating the input array X as a circular array due to its periodicity. Finally, the weights are frozen at the final values obtained and the NSMP input samples are shifted through the transversal filter (with the LMS algorithm inactive) to form the array of equalizer output samples. The mean square error is computed based on these output samples.

Finally, the change in noise energy is computed. This is based on the assumption that white Gaussian noise of unit energy is present at the equalizer input. The resultant quantity PWRN represents the noise energy that would be present after noise filtering due to the equalizer followed by the integrate-and-dump filter. The tap weights are converted to the corresponding filter frequency response by the FFT and the energy spectral response of the equalizer is determined. This spectrum is then multiplied by the $(\sin x/x)^2$ spectrum of the ideal integrate-and-dump detector.

In program SIMA, subroutine ADEQ follows the automatic gain control signal normalization and the receiver filter and precedes the phase tracking and bit synchronization loops. Thus, the equalizer is located in the IF section of the receiver.

3.9   CONVOLUTIONAL ENCODING AND VITERBI DECODING

Coding is performed in subroutine XCOD and Viterbi decoding is under the control of subroutines VDEC and VITERB. Parameters and usage of these subroutines are given in Appendix A.

The PN data sequences are generated as before and used as input bits to the convolutional encoder. If the parallel coding mode is used, the two PN sequences are independently encoded, one for the I channel and one for the Q channel. In the serial mode, only one PN sequence is used and the coded symbols are alternately used for the I and Q channels. For R = 1/2 coding, each input PN bit generates two coded symbols. The first of the pair is used for the current I channel symbol and the second is used for the corresponding Q channel symbol. The shift register tap connections used for encoding the data are the following:

11001 and 10111 for K = 5

1101101 and 1001111 for K = 7

These coded bit streams are used as the quadriphase modulator input. For staggered quadriphase (SQPSK) operation, the Q channel symbols are delayed by one half the symbol time.

The modulated data can be passed through any combination of the system elements modelled in SIMA. This includes transmitter and receiver filters, amplifiers, phase tracking, bit synchronization, and adaptive equalizer. Either an integrate-and-dump or a sampling detector can be used to detect the coded bits. The detector outputs form the soft decision inputs to the Viterbi decoder.

In the program structure used, the original PN data bits are stored in an array PN for use in the decoding subroutine VDEC. To provide for decoder symbol sync and proper sync between decoder output bits and bits in array PN, subroutine CORD is called. CORD performs a crosscorrelation between the soft decision outputs and the original coded bits input to the modulator to permit the required sync alignment.

The detector outputs can be input to the decoder directly in analog form or they can be quantized to 8 levels using one of the 3-bit quantizers shown in Figures 3-13 and 3-14. The quantizer level spacings shown in Figure 3-13 are based on results of using dither control for adjusting the equalizer weights (Section 5.1.2). The level spacings of Figure 3-14 are derived from the "alpha flunk" information (Section 5.2.4) available on the QPSK/BPSK modem for monitoring modem performance. The levels assigned are based on the fact that the received signal level has been normalized to unity by the AGC action.

Depending on the mode of coding used, the (analog or quantized) soft decision quantities from the I and Q detectors are decoded by separate decoders (parallel mode) or alternately used as inputs to a single decoder (serial mode). A Monte Carlo simulation is used with Gaussian noise added to each input. Subroutine SNORM generates the random noise samples with zero mean and unit variance. For a specified received E_b/N_o, expressed in dB, these noise samples, after passing through the channel, are divided by the quantity

$$\sqrt{2 \cdot NSB \cdot 10^{.1E_b/N_o} \cdot R^{-1}}$$

3-35

776-3582
UNCLASSIFIED
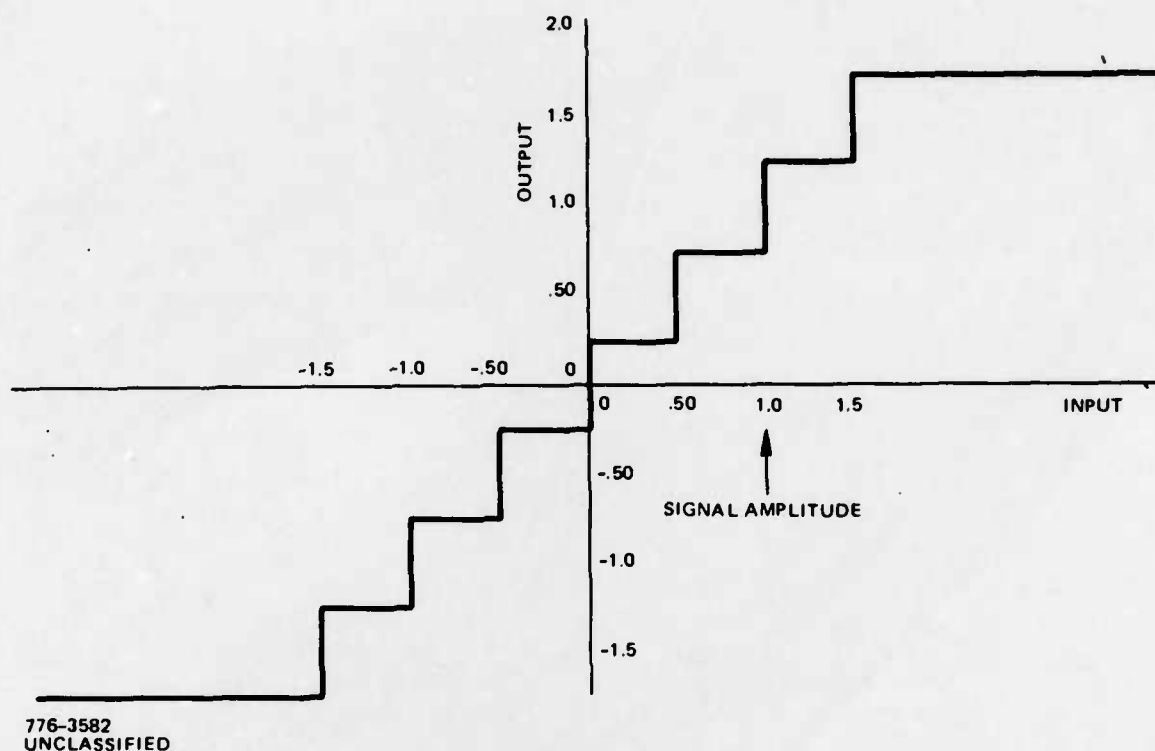
Figure 3-13. 3-Bit Quantization for Dither Control

776-3583
UNCLASSIFIED
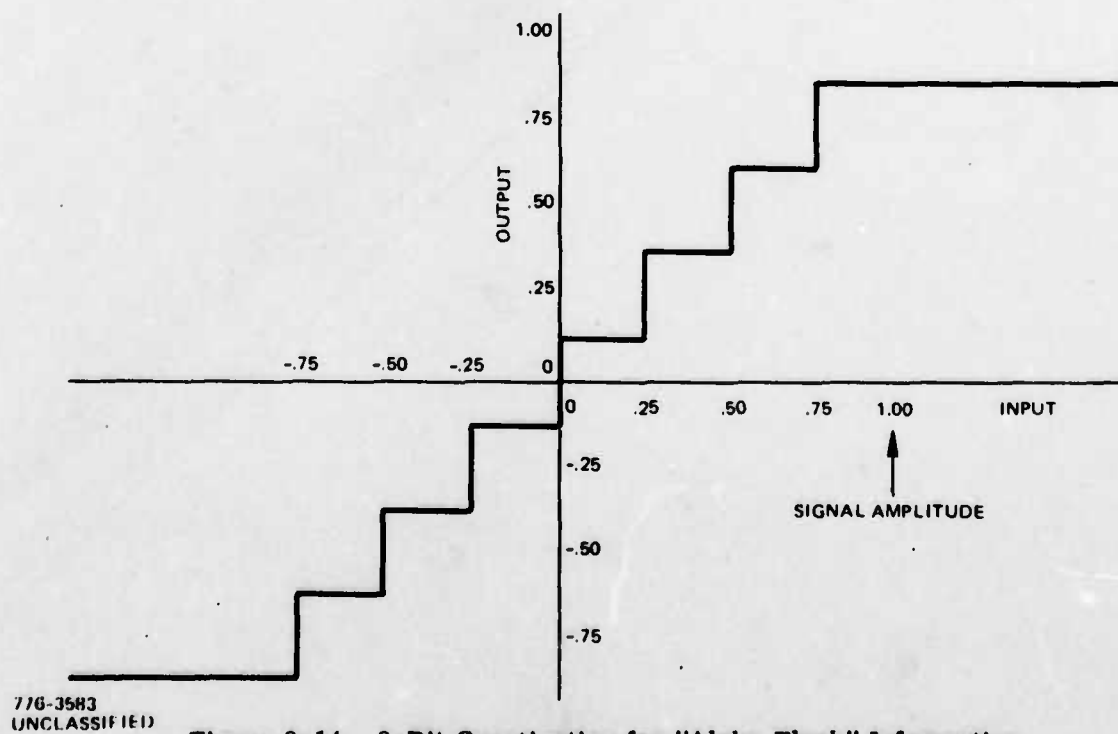
Figure 3-14. 3-Bit Quantization for "Alpha Flunk" Information

before being added to the soft decisions. The number of noisy decisions run through the decoder is given by parameter KNT, where KNT is selected to be at least ten times as great as the inverse of the expected decoder probability of bit error. An error is detected by comparing each decoded bit with the appropriate PN bit transmitted.

System degradation is measured by first determining the input E_b/N_o which would produce the same resulting probability of bit error in an ideal system using a convolutional encoder/Viterbi decoder with the same parameters. This (ideal) E_b/N_o is converted to decibels and subtracted from the actual receiver input E_b/N_o (in dB) used in the simulation run.

SECTION IV
ERROR CORRECTION CODING

Satellite communication channels are often power limited and the received signal-to-noise ratio limits the bit error rate that can be achieved. Convolutional coding and Viterbi decoding is often used on such channels to improve the system performance.

Coding introduces redundancy so that the transmitted symbol rate is greater than the information data bit rate, implying a bandwidth expansion. On the other hand, coding serves to reduce the E_b/N_o required at the receiver for a given bit error rate. Rate -1/2 convolutional encoding and soft decision Viterbi decoding can provide a 4-6 dB improvement in E_b/N_o in a relatively low cost manner, but at the cost of doubling the bandwidth to accommodate the coded bits. The tradeoff between bandwidth and E_b/N_o depends on whether the channel is power limited or bandwidth limited. If the channel is power limited and the increased bandwidth requirement can be tolerated, coding provides a more economical method of achieving a given bit error rate than increasing transmitter power and/or antenna gains.

Note that in this report, data bit rate means the rate of information bits into the convolutional encoder or out of the Viterbi decoder. Symbol rate signifies the transmitted bit rate (coded bit rate) over the channel; i.e., the bit rate from the encoder output to the decoder input.

Most investigations of Viterbi coding evaluate the performance on memoryless channels having additive white Gaussian noise[1, 2]. The current study determines the effects of system distortions and multiple accessing signals on decoder performance. A Monte Carlo computer simulation is used to model the Viterbi decoder operation. In addition, two techniques for rapidly estimating coding degradation without resort to a Monte Carlo simulation are discussed. One is based on the uncoded system degradation at low E_b/N_o. The second method obtains the minimum metric difference as a measure of the asymptotic decoder performance.

A simple technique is presented for converting a rate-1/2 convolutional code to a rate-3/4 code by deletion (puncturing). A search was performed for constraint lengths of 5 and 7 to determine the best code and deletions for this mode of operation, which improves bandwidth utilization by 50 percent. The codes found are transparent and have free distance equal to the maximum possible for their constraint length. Simulations indicate an increase of about 1 dB in E_b/N_o over a rate-1/2 code of the same constraint length on the ideal coherent PSK channel.

4.1   DISCUSSION OF CONVOLUTIONAL ENCODING/VITERBI DECODING

This section covers basic aspects of the coding-decoding process. Further details of convolutional encoding and Viterbi decoding can be found in references 1 and 2.

4.1.1   CONVOLUTIONAL ENCODER

A convolutional encoder is a linear finite-state machine consisting of a K-stage binary shift register and n mod-2 adders. Each of the adders is connected to certain of the shift register stages and the pattern of connections specifies the code generated.

The input information data bits are shifted through the register b bits at a time. Following each b bit shift, the mod-2 adder outputs are sampled sequentially to yield the coded bits (symbols). These coded bits are input to the modulator. For each set of b information bits, n coded bits are generated. The code rate $R = b/n$, where $b < n$. For $b = 1$ and $n = 2$, a $R = 1/2$ code is produced.

The constraint length of the code, **K**, may be defined as the number of shifts over which a single information bit influences the decoder output. Then, the state of the encoder is the contents of the $K - b$ shift register stages. The n output symbols are determined by the current encoder state and the next b input information bits.

Since the performance of the decoder for memoryless symmetric channels with white Gaussian noise depends on the (Hamming) distance property of the

convolutional code, * these distances are of importance. There is no loss of generality in computing the distance from the all zeroes codeword to all the other codewords for this set of distances is the same as the set of distances from any specified codeword to all the others. The "free distance" d_{free} is determined by the word(s) with minimum distance. d_{free} characterizes the asymptotic probability of error performance of the decoder. [1]

A systematic convolutional code refers to a code for which one of the coded bits is equal to the current input data bit. A systematic code has smaller d_{free} than a corresponding nonsystematic code for the same constraint length. For large K and R = 1/2, systematic codes have the performance of nonsystematic codes of half the constraint length. As an example, for K = 5 a rate-1/2 systematic code has d_{free} = 5 while the rate-1/2 nonsystematic code has d_{free} = 7.

4.1.2 MODULATION

The coded binary data is used to modulate the RF carrier. When quadriphase (QPSK) or staggered quadriphase (SQPSK) modulation is assumed, the quadrature channels can be independently (parallel) encoded with different data sources for the inphase and quadrature components or a single encoded data source can be used to (serially) modulate the two channels.

For R = 1/2 encoding and parallel modulation, the two encoded bits generated for each data bit are used consecutively to modulate either the in-phase or the quadrature channel. For R = 1/2 encoding and serial modulation, one of the pair of encoded bits produced for each data bit modulates the in-phase channel while the second bit modulates the quadrature channel. The overall data rate is the same for either case.

The signal-to-noise ratio at the receiver input assumes the presence of white Gaussian noise with one-sided noise spectral density N_o. The signal power present is P, giving a signal power-to-noise density = P/N_o. For an information rate R_D in bits per second, the received energy per bit to noise density ratio is given by

---

*The number of transmitted bits in which the correct codeword differs from any incorrect word which finally remerges with the correct word.

$$(E_b/N_o) = (P/N_o) \cdot (1/R_D) \qquad (1)$$

or

$$R_D = (P/N_o)/(E_b/N_o) \qquad (2)$$

At the receiver, the signal plus added white Gaussian noise is phase tracked and bit sync is established. An integrate-and-dump matched filter detector is assumed. The matched filter output forms the soft decision input to the Viterbi decoder. If quantization of the matched filter output were used, some degradation in E_b/N_o necessary to maintain a specified error rate would result. This varies from less than .25 dB for 8-level quantization to about 2 dB for hard limiting (2-level) of the output.

4.1.3   VITERBI DECODING

On a memoryless binary symmetric channel (BSC), errors which transform a code symbol from -1 to +1 or +1 to -1 are assumed to occur independently from symbol to symbol with probability p. If all message sequences are equally likely, the decoder which minimizes the overall probability of error is one which examines the distorted received sequence and chooses the data sequence corresponding to the transmitted code sequence which is closest to the received sequence in Hamming distance; i.e., the transmitted sequence which differs from the received sequence in the minimum number of symbols.

Viterbi decoding reduces the effort required for maximum likelihood sequence decoding by taking advantage of the code tree structure in which the paths remerge after the constraint length. The decoder has a finite number of states, and the minimum distance path of the paths merging into a given state at a given node is referred to as the survivor. Thus, at each node, it is necessary only to remember which was the minimum distance path from the received sequence for each decoder state, as well as the value of that minimum distance.

By this technique, we proceed through the received sequence and at each step for each state preserve one surviving path and its distance (metric) from the received sequence. If, in a given comparison between merging paths, the metrics are equal we randomize the decision (flip a coin) to determine the survivor. Channel distortion perturbs the soft decisions and thereby perturbs the metric differences.

An infinitely long path history cannot be stored. Therefore, an important consideration is the effect of path memory truncation. If a fixed amount of path history storage is provided, the decoder can output the oldest bit corresponding to that path (one for each state) with the highest metric M branches forward. If M is several times as large as K, it can be shown that the additional bit errors introduced by the truncation are very few; also, the output can be taken from any path.

Block synchronization is not required for convolutional codes as is needed for block codes. Synchronization will always be achieved and generally takes no more than 3 or 4 constraint lengths of received bits. Code symbol sync within a branch is required. For a R = 1/2 code, the two-way ambiguity as to where each two code-symbol branch begins must be resolved. This is called node synchronization. The wrong synchronization selection can be detected because the mismatch will cause all the path metrics to be small since there will not be any correct path in this case.

4.1.4   COMPUTER SIMULATION RESULTS

Several cases were run to validate the program, as described in Section 3.9, and to evaluate the performance of coding for ideal and degraded channels. Additional cases of coded channels are given in following sections of this report. Due to the time-consuming nature of the Monte Carlo simulation runs, most were run for an E_b/N_o of 4 dB. This E_b/N_o is in the range for which coding is most often used.

The following coding parameters were used (unless otherwise specified):

> Coding Mode = Serial
> Rate(R) = 1/2
> Constraint Length(K) = 5
> Path Delay(M) = 3 · K
> Modulation Format = Staggered Quadriphase (SQPSK)

The number of decoded bits run in the Monte Carlo simulation included at least ten bit errors. For higher error rates, at least 100000 decoded bits (200000 soft decisions) were run.

The first set of runs assumed an ideal SQPSK channel. This run served to validate correct performance of the decoder simulation program. Both the serial and parallel coding modes were used. The results are plotted in Figure 4-1 for an

PROBABILITY OF BIT ERROR (y-axis: 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5})

E_b/N_o (dB) (x-axis: 0 to 8)

776-3282
UNCLASSIFIED
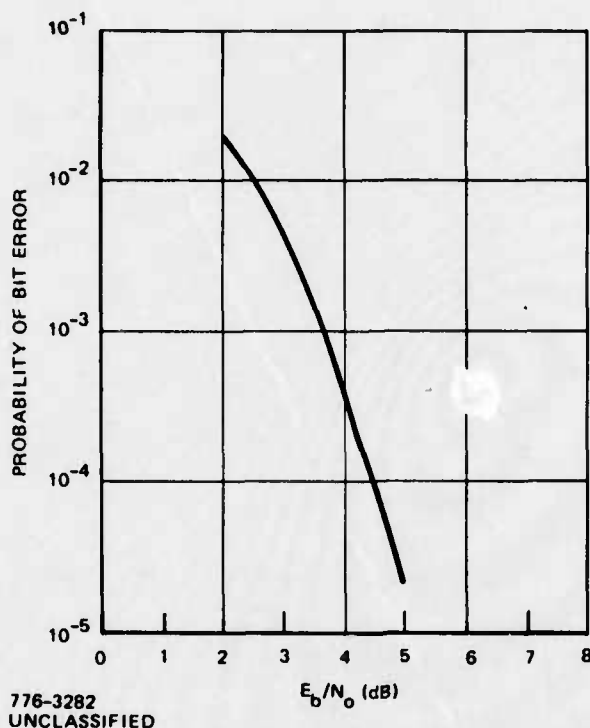
Figure 4-1. Performance of Coding Over Ideal Channel
(R = 1/2, K = 5, M = 15, Mode = Serial)

E_b/N_o range of 2 dB to 5 dB. Both coding modes produced essentially the same results
so only one curve is shown. The results are in excellent agreement with both
theoretical predictions and the simulation results of reference 2. This curve is used
throughout this study as the benchmark for computing performance degradation for
non-ideal channels.

Figure 4-2 shows the performance for a range of E_b/N_o for closely spaced
multiple accessing signals in the channel. The center channel, for which the error
rate was determined, is surrounded by two adjacent equal-amplitude (not necessarily
the same amplitude as the center channel) SQPSK signals. All three channels have the
same bit rate but different PN sequences. The outer channels have bit transitions
delayed one quarter bit from the transitions of the center channel.

4-6
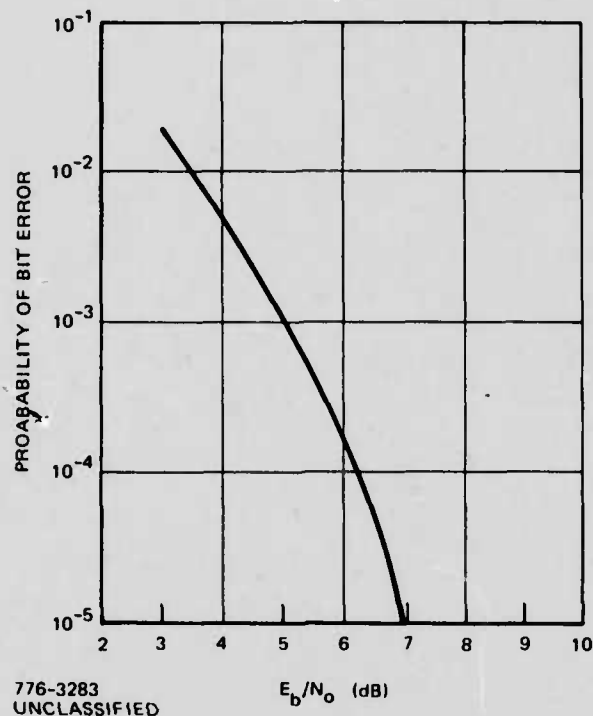
Figure 4-2. Performance of Coding With Filtered Adjacent Channel
Signals Each 20 dB Above Center Channel. Channel
Spacing FCENT = $2/T_b$. Coding Parameters: R = 1/2,
K = 5, M = 15, Mode = Serial.

The outer two channels are symmetrically spaced about the center channel
with center frequencies of $\pm 2/T_b$, where T_b^{-1} is the transmitted symbol rate in the
in-phase or quadrature channel. Thus, the first null frequencies of the center and each
outer channel are coincident. The outer channels are 20 dB stronger than the center
channel. All channels are linear with each channel bandlimited by transmitter
filters and the receiver has a filter centered on the middle channel. All filters have
a one-sided bandwidth B extending to the first null frequency ($BT_b = 1.$) and are five
pole filters with Butterworth amplitude response and zero phase shift for all fre-
quencies. Without coding, a performance degradation of 3.18 dB results at E_b/N_o =
10 dB and 1.08 dB is incurred at E_b/N_o = 2 dB. With coding, the asymptotic
degradation (described in Section 4.2) predicted by the minimum metric is 2.33 dB.
The coding simulation produced the curve shown in Figure 4-2.

Table 4-I gives a variety of simulation results for several different forms
of channel distortion. The first column gives the channel distortion parameters. The
second column gives the coding parameters. The third column gives the received

4-7

Table 4-I. Degradations Produced for Coded and Uncoded Systems for Various Channel Distortions

System Distortion Parameters	Coding Parameters	E_b/N_o	Viterbi Simulation	Asymptotic Degradation	Uncoded System $E_b/N_o = 2$ dB	$E_b/N_o = 10$ dB
Sinusoidal (SK = 30°) Filter (BT$_b$ = 1.)	Serial Mode K = 5	6	1.8	3.2	2.15	5.09
Sinusoidal (SK = 45°) Filter (BT$_b$ = 1.)	Serial Mode K = 5	8	3.7	5.12	3.91	9.81
Sinusoidal (SK = 45°) Filter (BT$_b$ = 1.) Equalizer (NTAP = 20, ISP = 16)	Serial Mode K = 5	4	.73	1.3	.68	.87
Sinusoidal (SK = 45°) Equalizer (NTAP = 20, ISP = 16)	Serial Mode K = 5	4	.18	.19	.21	.27
Hughes' TWT (BO = 0 dB) 3 Accessing Signals	Serial Mode K = 5	4	.62	1.60	1.13	3.21
Hughes' TWT (BO = 0 dB) 3 Accessing Signals	Parallel Mode K = 5	4	.65	1.72	1.13	3.21

NOTE: E_b/N_o and degradations are given in decibels. Evaluation of asymptotic degradation is discussed in Section 4.2.

E_b/N_o at which decoding was performed. The fourth column gives the degradation that results from the Monte Carlo simulation. The fifth column tabulates the corresponding asymptotic coding degradation predicted by the minimum metric technique (Section 4.2) as $E_b/N_o \rightarrow \infty$. The last two columns give the degradations produced for the equivalent uncoded SQPSK channels for $E_b/N_o = 2$ dB and $E_b/N_o = 10$ dB, respectively.

The first set of runs assumed sinusoidal phase distortion in the channel. The phase distortion models and parameters are discussed in Section 3.4. In all cases, one cycle of ripple takes place over the frequency range from band center to the first data null (FR = 1.). Peak phase errors of 30° and 45° were assumed.

A run was performed to evaluate the performance improvement with coded data resulting from use of IF adaptive equalization in the receiver.

The 32 received symbols, sampled 32 times per bit, are stored in an array X of size 1024. The 1024 samples of array X are shifted through the transversal filter portion of the equalizer with each shift generating an output time sample S. No noise is added to the input samples at this point. The equalizer adapts over a total of $12 \cdot 1024$ samples by treating the input array X as a circular array due to its periodicity. Finally, the tap weights are frozen at the final values obtained and the 1024 input samples are shifted through the transversal filter (with the LMS algorithm inactive) to form the array of equalizer output samples.

The random noise samples (obtained from SNORM) are passed through the equalizer using the final set of tap weights obtained. The resulting output noise samples are correlated to some extent and the noise power is changed due to the transversal filter action. These output noise samples are appropriately scaled by the E_b/N_o value used and added to the detected soft decisions prior to the decoder input.

Results 'n Section 6.2 show a great improvement with equalization for uncoded data based on hard decisions. This simulation is intended to show the corresponding system performance using soft decision outputs to the decoder based upon sinusoidal distortion with a peak phase error of 45°. Degradation was measured both with and without an ideal 5-pole Butterworth filter ($BT_b = 1.$) in the transmitter.

Another set of runs evaluated the degradation due to nonlinear amplification by a Hughes' TWT. Three accessing SQPSK signals passed through the TWT. The frequency spacing between the channels was π/T_b (Hz) and the decoded error rate is based on the center channel. No channel filtering was employed. A backoff (BO) of 0 dB was employed. To determine whether the AM-PM conversion would affect the

parallel coding mode differently than the serial mode, both were simulated. No significant differences result from the mode used.

4.1.5   CONCLUSIONS

The Viterbi runs illustrate several important results. First, the Monte Carlo simulation results show that the degradation due to system distortions is close to that predicted by the degradation at E_b/N_o = 2 dB for an uncoded system. In addition, the asymptotic degradation, as predicted by the minimum metric technique of Section 4.2 is an upper bound for all cases tested. Both techniques yield results in a far more economical manner than a Monte Carlo simulation and can be used to get an estimate of the degradation to convolutionally encoded/Viterbi decoded signals.

Coding does not appear to improve the tolerance for phase distortion even for a fixed transmitted bit rate, where the user-to-user data rate is reduced by the coding. Fortunately, the use of adaptive equalization of the channel works effectively for soft decision decoding and greatly lowers the degradation produced by channel phase distortion. This is not surprising since the adaptive LMS algorithm used minimizes the rms error between the desired channel response and the equalizer output response.

For three signals accessing the channel (with the outer channels 20 dB stronger than the center channel) and a channel spacing of $2/T_b$, where T_b is the duration of the transmitted symbol (on in-phase or quadrature) the degradation is 1.2 dB over that of the ideal channel. For the equivalent uncoded three signal case with a channel spacing of $2/T_d$, where T_d^{-1} is the transmitted data bit rate, the degradation is 3.18 dB over the ideal channel. While the degradation is less for the coded system for such null-to-null spacing it must be remembered that $T_d = 2T_b$ so that the channel separation used for the coded channel is twice that used for the equivalent uncoded channel.

For three accessing signals, coding shows far less degradation due to the effects of nonlinear amplification and AM-PM conversion than the corresponding uncoded system. For the Hughes' TWT at 0 dB backoff, the uncoded system produced 3.21 dB degradation while the coded system has only .62 dB degradation.

These results indicate that coding improves channels as long as the hard decision bit error rate is not severe (as in the case of sinusoidal phase distortion with large peak phase) and the reduced user-to-user data rate can be tolerated. Therefore, while coding does not conserve the spectrum (in fact, just the opposite), it is highly desirable where power must be conserved (since coding reduces the required E_b/N_o) and channel distortion is not severe.

4.2   TECHNIQUES FOR ESTIMATING THE PERFORMANCE OF CODED SYSTEMS

On the memoryless channel with additive Gaussian noise, decoder performance, P_e vs E_b/N_o, is established by the distance properties of the convolutional code; i.e., the number of transmitted bits in which the correct code word differs from any incorrect word which finally remerges with the correct one[1]. Summing the pairwise probabilities of error for the decisions between the correct code word and all incorrect words yields an upper (union) bound on P_e. As E_b/N_o increases, the dominant contribution to the bound is from the word(s) with minimum distance; hence, this "free distance" d_{free} characterizes the asymptotic probability of error performance of the decoder as $E_b/N_o \rightarrow \infty$.

The best[3] constraint length K = 5, rate - 1/2 code has d_{free} = 7; hence, the asymptotic improvement with respect to the uncoded channel is by the factor 7/2, or 5.4 dB, as $E_b/N_o \rightarrow \infty$. The best constraint length K = 7, rate 1/2 code has d_{free} = 10, giving an asymptotic improvement by 10/2, or 7.0 dB. *

The actual performance degradation caused by channel distortion may be measured at any E_b/N_o by a Monte Carlo simulation transmitting random data. For simulation convenience in computing the intersymbol interference, however, a periodic pseudorandom data sequence is actually transmitted repetitively** until a satisfactory error rate measurement has been made.

---

*The actual coding gain realized at $P_e = 10^{-5}$ is less than this, of course.

**An incorrect code word differing by d_{free} must have diverged from the correct word at least one constraint length K earlier. In fact, we must allow for a time on the order of K^2, according to Bahl and Jelinek[4]. This sets a lower bound on the period of the repetitive data sequence.

An asymptotic evaluation of coding gain is only an approximation, useful for the range of E_b/N_o where a computer simulation is not practical because of the large number of bits required to obtain statistically valid measurements as $P_e \to 0$.

4.2.1 ESTIMATING DECODER PERFORMANCE ON CHANNEL WITH INTER-SYMBOL INTERFERENCE

To understand how intersymbol interference degrades decoder performance, we first review briefly the internal operation of the decoder. Fundamentally, the decoder correlates the received sequence of soft decisions with all possible code words, and finally selects the code word yielding maximum correlation. The accumulated correlation values are termed the "metrics". Since the correct code word should have the highest metric (otherwise, the decoder makes an error), the metric differences when the correct word is compared to the incorrect words in the absence of noise characterize the ability of the decoder to select the correct code word at low E_b/N_o.

Channel distortion perturbs the soft decisions, and thereby perturbs the metric differences in the absence of noise, some being increased, others decreased. Since a decoder error due to noise is most likely to occur from a comparison of code words for which the metric difference is small in the absence of noise, the asymptotic performance of the decoder can be estimated from the minimum metric difference.

The above discussion leads to a simple evaluation of the asymptotic degradation due to channel distortion, when error correction coding is employed. The minimum metric difference is found over the period of the pseudorandom data sequence. Its reduction relative to the minimum metric difference for the ideal channel determines the asymptotic degradation in E_b/N_o. For a code with free distance d_{free}, the minimum metric difference on the ideal channel with unity bit amplitudes is $2 d_{free}$, since an incorrect word differs from the correct word in at least d_{free} transmitted bits. Note, this procedure presumes white Gaussian receiver noise which produces an independent additive Gaussian random variable on each soft decision. Also, the minimum metric difference is presumed to occur for an incorrect word differing from the correct word in d_{free} transmitted bits. Furthermore, unquantized soft decisions are presumed.

If the receiver incorporates a bandpass filter or an equalizer, there is an additional approximation of neglecting correlation between the noise samples on adjacent soft decisions, so that the noise variance on a soft decision is still multiplied by d_{free} to get the noise variance for the pairwise decision between code words differing in d_{free} bits.

4.2.2 MODIFICATION TO VITERBI ALGORITHM TO FIND THE MINIMUM METRIC DIFFERENCE

The minimum metric difference required to characterize the asymptotic decoder behavior is not normally an output from the Viterbi algorithm. The rate-1/2 Viterbi decoder of constraint length K has storage for 2^{K-1} internal states, and a metric value is stored for each state. As each successive node (pair of transmitted bits for a rate-1/2 code) is received, the 2^{K-1} metric values are updated. In particular, for each state a comparison is made between the two possible code words leading to that state, and only the one producing the higher metric value survives. The lower metric value and corresponding code word are simply discarded. A decoder error event occurs at a given node in the presence of noise whenever the correct code word fails to survive at that node due to the effect of the noise.

The procedure for modifying the Viterbi algorithm to find the minimum metric difference, then, is to observe the difference in the pair of metric values computed for updating the correct state at each successive node in the absence of noise. Over the succession of correct states, the minimum of this difference is retained and finally read out.

A problem becomes apparent at this point; namely, the Viterbi algorithm does not actually know the sequence of correct states as the metric differences are computed, and only selects the sequence of output data bits after a large internal delay. However, provided that the decoder does not make errors due to the channel distortion in the absence of noise, the sequence of correct states can be reconstructed from the output bits, except that the reconstructed sequence will be delayed by the internal memory. For the case where the data sequence is periodic, as described above, the sequence of correct states is also periodic. Hence, the path memory length can be adjusted so that the reconstructed sequence of correct states is delayed by exactly one period, therefore, is effectively undelayed. In this way, the metric difference leading to the correct state can be selected at each successive node.

4.2.3 RESULTS OF COMPUTER SIMULATION

The previously described modification of the Viterbi algorithm was implemented and used to determine system performance with a variety of channel impairments. The degradations investigated are the effects of a system filter, system phase distortion (sine, cosine, and parabolic), and adjacent channel interference.

All runs were performed at essentially an infinite signal-to-noise ratio to determine the asymptotic performance. Two constraint lengths were used, K = 5 and K = 7, to determine the difference in performance for the various cases simulated. Rate-1/2 coding was used throughout. The minimum metric was measured following the first period of the encoded sequence to ensure that start-up transient effects are nonexistent.

The degradation, in decibels, is measured by taking the ratio of the minimum metric difference obtained to the metric difference which results when decoding the undistorted data sequence. This metric difference = 14 for K = 5 and = 20 for K = 7 in the absence of distortion. Table 4-II shows the degradation for a variety of sinusoidal, cosinusoidal, and parabolic channel phase distortions. In all cases the transmitter contained a single five pole Butterworth filter with BT_b = 1. The degradation due solely to the transmit filter is the first entry in Table 4-II. The degradation obtained can be compared with the resulting degradation for the same distortions in an uncoded system as given in Table 4-II for an input E_b/N_o = 10 dB and the same transmitted bit rate (i.e., twice the user-to-user data rate).

The results show that the asymptotic degradation falls between the uncoded degradations at E_b/N_o = 2 dB and E_b/N_o = 10 dB as given in Table 6-I. Thus, coding does not appear to improve the tolerance for phase distortion even for a fixed transmitted bit rate, where the user-to-user data rate is reduced by the coding.

The procedure used for computing the minimum metric difference is valid unless an error is made, indicating that the decoder selected an incorrect state. For cosinusoidal distortion with 45° peak phase, an error was made for the case K = 5. Therefore, an x was entered in Table 4-II for that case. All other simulation runs in Tables 4-II and 4-III produced zero errors in the absence of noise.

Table 4-III shows the degradation resulting from two adjacent signals as a function of their frequency separation and the system filtering used. Both adjacent channels are symmetrically located about the center channel and the system is assumed to be linear. Channel spacing is normalized to the transmitted bit rate on the inphase or quadriphase channel. Thus, a spacing of 1.0 implies that the adjacent channels are located at the first nulls of the center channel spectra. All channels have different PN sequences. The transmitted bit rate is the same for both coded and uncoded operation; hence, the user-to-user data rate is halved when a rate-1/2 code is used.

Table 4-II. Degradations Computed From Minimum Metric Difference for Transmit Filter and Channel Phase Distortion

Phase Distortion	Parameters (SK, FR)	Degradation (dB)		
		Uncoded*	K = 5	K = 7
None	0°, 1	.78	.88	.86
Sinusoidal	15°, 1	2.24	1.87	1.83
	30°, 1	5.09	3.2	3.55
	45°, 1	9.81	5.12	7.51
Cosinusoidal	15°, 1	2.08	1.53	1.63
	30°, 1	4.58	3.08	2.92
	45°, 1	7.65	x	5.28
Parabolic	45°, 1	1.28	1.12	1.25
	75°, 1	1.42	1.26	1.39
	90°, 1	1.60	1.37	1.53
	135°, 1	2.18	1.85	1.96
	180°, 1	3.05	2.67	2.5

Table 4-III. Degradations Computed From Minimum Metric Difference as a Function of Adjacent Channel Spacing and System Filtering

System Parameters	Channel Spacing	Degradation (dB)		
		Uncoded*	K = 5	K = 7
P_1 = 0 dB	5.0	.12	.88	.86
P_2 = 0 dB	4.0	.41	.88	.87
P_3 = 0 dB	3.0	.88	1.31	1.25
	2.5	.92	1.66	1.51
No XMT Filter	2.0	.98	2.16	1.95
No RCV Filter	1.75	1.20	2.84	2.57
	1.5	1.50	3.6	2.91
P_1 = 0 dB	5.0	.40	1.17	1.16
P_2 = 0 dB	4.0	.42	1.17	1.17
P_3 = 0 dB	3.0	.45	1.20	1.19
	2.5	.49	1.27	1.25
XMT Filter	2.0	.59	1.33	1.30
RCV Filter	1.75	1.22	1.38	1.33
	1.5	1.64	1.73	1.55
	5.0	.41	1.18	1.16
P_1 = 20 dB	4.0	.43	1.18	1.17
P_2 = 0 dB	3.0	.48	1.22	1.18
P_3 = 20 dB	2.5	.58	1.29	1.19
	2.0	3.18	2.33	2.00
XMT Filter	1.75	9.73	8.92	6.2
RCV Filter	1.5	--	23.9	11.6

*Computed for an input E_b/N_o = 10 dB.

4-15

The first set of results assume that all three channels have identical power and no bandlimiting takes place in the system. The second set of runs assume that all three channels have identical power but all three channels have been band-limited in the transmitter and a receiver filter is also used. The third set of results assumes both transmit and receiver filtering but the adjacent channels are now 20 dB stronger than the center channel.

In all cases, the filters modelled had the amplitude response of a 5-pole Butterworth filter (BT = 1.) with zero phase shift, as described in Section 3.2.7. As shown in Table 4-III, both transmit and receive filters are necessary to permit close spacing of the channels without significant degradation, when the power levels are uncontrolled. Results comparing the estimates with actual Monte Carlo simulation results are given in Table 4-I and in later sections of this report.

4.3 VITERBI DECODER PERFORMANCE WITH 3-BIT QUANTIZED INPUTS

The previous evaluations of Viterbi decoder performance, assumed no quantization of the soft decision decoder inputs. The typical QPSK/BPSK modem, however, quantizes the soft decisions to 3 bits. Coded systems that use receiver outputs quantized to more than two levels require an A/D converter with thresholds based on the noise variance. Since the quantizer level settings are controlled by the automatic gain control (AGC) circuitry in the modem, the sensitivity of performance with respect to the level settings is of prime importance. Previous studies of Viterbi decoder performance[2] have shown that the uniform quantizing levels should be spaced by approximately 0.5, where the standard deviation of the noise is unity. Fortunately, decoding performance was shown to be relatively insensitive for level spacings varying from this amount over an octave range.

For evaluation of dither control of adaptive equalizer weights with quantized modem outputs (see Section 5.2.2.4), a spacing was used as shown in Figure 3-13. Results obtained indicated that good equalization is obtained when controlled by such quantized modem outputs.

An alternate 3-bit quantizer is used in the QPSK/BPSK modem to obtain "alpha flunk" information. The purpose of the alpha flunk output is for monitoring the modem performance. The level settings of this quantizer are shown in Figure 3-14. As can be seen, the quantizing levels are spaced by 0.25, where the signal amplitude is 1.0. The "alpha flunk" measurement is based on the middle bit, which

indicates a soft decision amplitude less than 0.5. The same information can be obtained from the "dither" quantizer of Figure 3-13 by use of the two magnitude bits. The 00 case corresponds to detected outputs falling in the quantizer region extending from [-0.5, 0.5]. A single AND-gate can be used to obtain the "alpha flunk" information from the "dither" quantizer output.

Both quantizers are used in this investigation to determine the performance degradation resulting for the Viterbi decoder accepting these inputs as opposed to unquantized soft decision inputs.

4.3.1 SIMULATION RESULTS

Program SIMA was modified to permit insertion of the 3-bit quantizers. The integrate-and-dump detector outputs with additive noise were quantized to 3 bits, and these quantizer outputs were used as the soft decision decoder inputs. Prior to detection, AGC is used to normalize the desired signal power at the receiver input to unity. The quantizer desired is selected by the user.

Figure 4-3 shows the Monte Carlo simulation results. An ideal channel

776-3284
UNCLASSIFIED
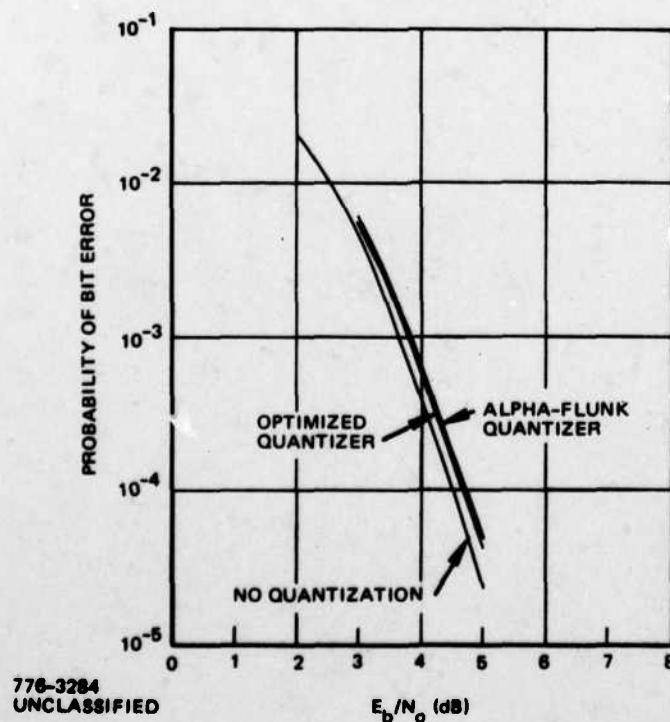
Figure 4-3. Performance of Viterbi Decoder on Ideal Channel With Unquantized Inputs and Inputs Quantized to 3 Bits With Two Different Level Spacings

4-17

was assumed with staggered quadriphase modulation. Rate-1/2 coding was employed and the constraint length is five. Virtually identical results were obtained for both serial and parallel coding modes. The best performance was obtained for the case of unquantized inputs, as expected. The performances of the two quantizers are similar with the alpha flunk quantizer being slightly inferior. The conclusion of this investigation is that satisfactory Viterbi decoder performance can be obtained with either quantizer.

4.4   AN APPROACH TO RATE-3/4 CONVOLUTIONAL CODING/VITERBI DECODING

To obtain an improved bandwidth utilization for coded data, a rate-3/4 code can be employed. The benefit of a reduced E_b/N_o and a greater tolerance for intermodulation distortion is achieved, compared with uncoded data, without as great a reduction in data rate as with a rate-1/2 code.

Rate-3/4 convolutional encoding may be described as a process which inputs 3 data bits and generates 4 coded bits (symbols), where each symbol is defined by a parity sum over the recent history of the data bits (i.e., the constraint length). Figure 4-4 shows such a rate-3/4 encoder which has a constraint length of 9 data bits and a free distance equal to 6. The free distance is defined as the minimum distance between code words of arbitrary duration, and is a measure of code strength. The code in Figure 4-4 is the best possible for the given constraint length. [5]

Viterbi decoding is dependent on being able to identify a finite number of possible states of the encoder; for Figure 4-4, there are 2^6 states specified by the 6 data bits preceding the present 3 data bits. In other words, the 4 symbols are completely specified by the encoder state and the present 3 data bits. The decoder functions to find the sequence of states which is most likely to have produced the received symbols.

A rate-1/2 convolutional encoder generates 2 symbols for each data bit. A rate-1/2 encoder of constraint length 7 has 2^6 states; i.e., the 2 symbols are specified by the encoder state and the present data bit. The best rate-1/2 code of constraint length 7 has free distance equal to 10. [3]

On the ideal PSK channel with soft decision decoding, the asymptotic coding gain as $P_e \rightarrow 0$ equals (numerically) the product of the free distance and the code rate. This coding gain is commonly expressed in decibels. (The actual coding gain at a finite error rate is less.) Thus, the constraint length 7, rate-1/2 code has asymptotic coding gain equal to 7 dB, while the constraint length 9, rate-3/4 code has asymptotic coding gain equal to 6.5 dB. The Viterbi decoder for either code stores 2^6 states, hence, has roughly equivalent complexity for either code.
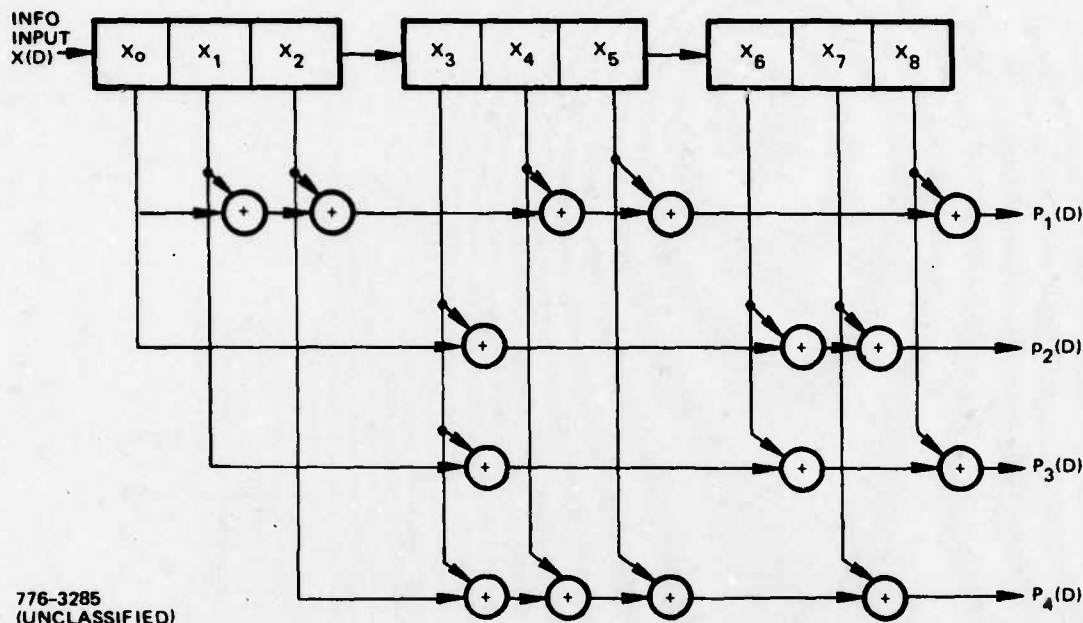
Figure 4-4. Constraint Length 9 Rate-3/4 Convolutional Code

4.4.1 CONVERTING A RATE-1/2 CODE INTO A RATE-3/4 CODE BY DELETIONS

Starting with a rate-1/2 convolutional code, we can obtain a rate-3/4 convolutional code by puncturing; i.e., deleting some of the symbols. Since the rate-1/2 encoder produces 6 symbols for each 3 data bits, we need to delete 2 of the 6. (Of course, the decoder must know which symbols have been deleted.) To accomplish this, a fixed deletion pattern is utilized on the 6 symbols produced from 3 data bits by the rate-1/2 encoder.

The rate-3/4 decoder must accept the symbols in blocks of 4, and this imposes a synchronization problem somewhat more complicated than for a rate-1/2 decoder, which accepts symbols in blocks of 2.

Let us number the 6 symbols from the rate-1/2 encoder in order as they appear from the two parity sums. Figure 4-5 illustrates this along with the three possible deletion patterns. Since the rate-1/2 convolutional encoder treats all data bits alike, all other deletion patterns can be reduced to one of the three indicated in Figure 4-5 by starting the numbering at a different bit and/or interchanging the two parity sums.

4-19

Symbols from First Parity Sum	1 3 5
Symbols from Second Parity Sum	2 4 6
Symbols from First Parity Sum	e 3 5
Symbols from Second Parity Sum	e 4 6
Symbols from First Parity Sum	e 3 5
Symbols from Second Parity Sum	2 e 6
Symbols from First Parity Sum	e e 5
Symbols from Second Parity Sum	2 4 6

Figure 4-5. Possible Deletion Patterns to Eliminate 2 Out
of 6 Symbols from Rate-1/2 Convolutional Code

The group property of the code is not affected by the deletions occuring
in a fixed pattern.

4.4.2 DECODING THE RATE-3/4 CONVOLUTIONAL CODE

The reason for taking a puncturing (i.e., deletion) approach to derive a
rate-3/4 convolutional code from a rate-1/2 code is that the structure of the soft-
decision Viterbi decoder for the rate-3/4 code can be identical to the rate-1/2
decoder. All that is necessary is to restore the deletions as erasures at the decoder
input; i.e., as zero amplitude* symbols so that they do not affect the metrics. The
data clock is 3/4 of the symbol clock for the rate-3/4 code, and the rate-1/2 decoder
input consists of pairs of symbols clocked at the data rate.

Functionally, the block of 4 symbols is reclocked to a block of 6 symbols,
two of which are erasures (zero amplitude symbols) in specified positions. As each
pair of these reclocked symbols is input to the decoder, a data bit outputs. Hence, 3
data bits result from the original 4 symbols.

4.4.3 OPTIMIZATION OF RATE-3/4 CONVOLUTIONAL CODE

The above technique for constructing a rate-3/4 convolutional code is
useful only if the resulting code is essentially the best possible. As mentioned above,
a measure of convolutional code capability is the free distance. The free distance
asymptotically determines decoder performance because the most likely decoding error
occurs to the incorrect word differing from the correct word in the least number of
symbols.

*With quantization of the soft decisions into the decoder, the erased symbols are
 restored to have the minimum amplitude.

By inputting a correct word (most conveniently the all-zeros word) the Viterbi decoder itself can be employed to find the minimum free distance. Each time the state metrics are updated, a comparison is made between the correct word and an incorrect word which remerges with the correct word, and only the survivor is retained. Effectively, a comparison is being made between the correct word and all possible incorrect words, but only one incorrect word, that which survived previous comparisons, is actually retained for the present comparison. This incorrect word has the minimum distance. Because the incorrect word previously diverged from the correct word, the metric difference for the comparison specifies the distance.

If the Viterbi decoder is started with all the state metrics set equal to zero, the remerging word did not necessarily diverge from the correct word. Hence, the decoder must be run for enough bits that the start up transient has disappeared; otherwise, the metric differences are too small, and the correct value for the free distance is not obtained.

The optimum rate-3/4 convolutional code which can be derived by deleting symbols of a rate-1/2 code can be found by an exhaustive search, performing this search for each of the three erasure patterns shown in Figure 4-5. For each code, the minimum distance after data bit number 48 and prior to data bit number 67 is recorded. However, the code is discarded if a decoding error is observed prior to data bit number 67, using a path memory length of 21 bits. The code and associated erasure pattern which maximizes the free distance is thereby found. This has been done* for constraint lengths equal to 5 and 7; for both, the best erasure pattern is to delete symbols 1 and 3. (This is the bottom pattern of erasures in Figure 4-5.) The resulting best rate-3/4 codes are transparent for both constraint lengths, even though the optimum rate-1/2 code for constraint length 5 is not transparent. Transparency is a desirable property, since synchronization is then easier in a PSK channel where the data can be received inverted.

The best codes for constraint lengths 5 and 7 are shown in Figure 4-6. These codes have free distance equal to the maximum possible for their constraint length. [5] However, if the best rate-1/2, constraint length 7 code is converted to rate-3/4 by deleting symbols, the resulting free distance is 5, erasing symbols 1 and 4. (This is the middle pattern of erasures in Figure 4-5.)

---

*The rate-1/2 Viterbi decoder program employed is constrained to use parity sums connected to the oldest data bit as well as the present data bit.

PARITY SUM → SAMPLE EVERY 3RD SYMBOL →

← DATA BITS   SYMBOLS

PARITY SUM →

FREE DISTANCE = 4
ASYMPTOTIC CODING GAIN = 4.8 dB

OPTIMUM CONSTRAINT LENGTH 5 CODE

PARITY SUM → SAMPLE EVERY 3RD SYMBOL →

← DATA BITS

PARITY SUM →

FREE DISTANCE = 6
ASYMPTOTIC CODING GAIN = 6.5 dB

OPTIMUM CONSTRAINT LENGTH 7 CODE
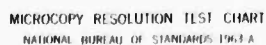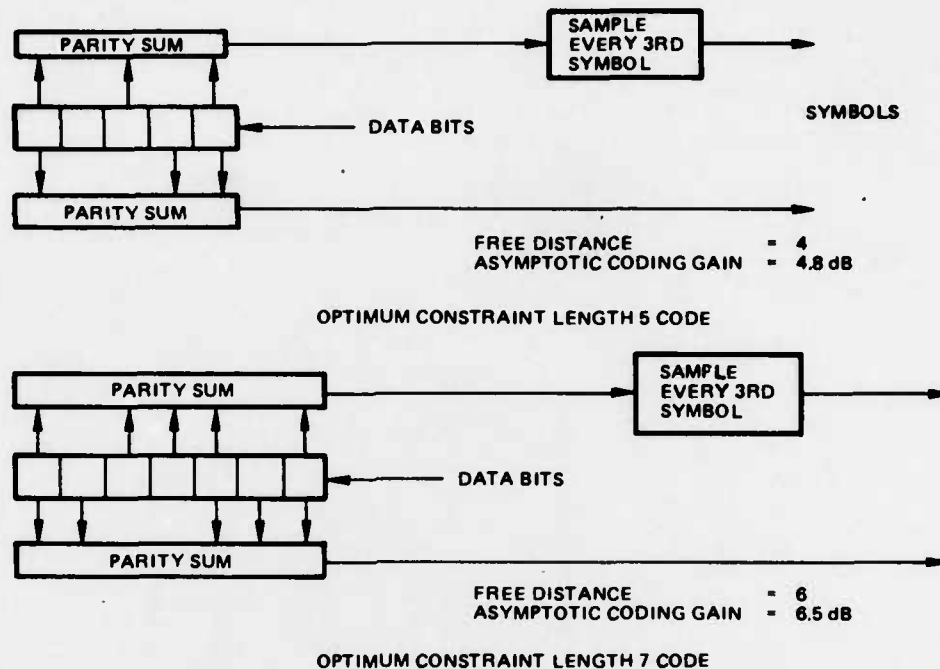
776-3286
(UNCLASSIFIED)

Figure 4-6. Rate-3/4 Convolutional Codes

The performance of the transparent, constraint length 7, rate-3/4 code has been obtained for unquantized soft decisions on the ideal coherent PSK channel by Monte Carlo simulation. At a decoded error rate of 2×10^{-4}, the simulation shows approximately 1 dB higher E_b/N_o is required for the rate-3/4 code than for the best constraint length 7, rate-1/2 code.

4.4.4   BPSK DATA TRANSMISSION

With a transparent code, the data bits may be differentially encoded prior to being convolutionally encoded. Then, an inversion over the BPSK channel due to 180° phase shift in carrier recovery at the receiver can be ignored. The output of the Viterbi decoder is, of course, differentially decoded to recover the data.

The 4 symbols in the block are transmitted serially. The decoder must obtain synchronization by trial until the correct symbol out of the four possibilities is identified as the start of the block.

4.4.5 QPSK DATA TRANSMISSION (SERIAL MODE)

With QPSK, the symbols are transmitted in pairs on the I and Q channels. (With staggered QPSK, the symbols are transmitted alternately on I and Q.) Now, the receiver can have swapped I and Q because of a 90° phase shift in carrier recovery at the receiver. However, inversion due to 180° phase shift can be ignored with a transparent code. Denoting the 4 symbols in the block as a b c d, the demodulator can output the patterns

$$
\begin{array}{ccccc}
I & ac & bd & ca & db \\
Q & bd & \overline{ac} & db & \overline{ca}
\end{array}
$$

where the overbar denotes inversion, and 180° shifts are ignored.

Thus, with QPSK, as with BPSK, there are four possibilities, one of which is to be identified as correct by trial.

4.4.6 CONCLUSIONS

A simple technique to generate a rate-3/4 convolutional code has been described. The encoder and decoder are rate-1/2, but some of the symbols (2 out of each 6) are deleted. The Viterbi decoder knows which symbols are deleted and restores them as erasures.

The best rate-3/4 codes generated in this manner achieve the maximum free distance possible for the given constraint length. The codes are transparent for both constraint length 5 and constraint length 7. Simulations indicate approximately 1 dB higher E_b/N_o is required on the ideal coherent PSK channel than for a rate-1/2 code of the same constraint length.

SECTION V
EQUALIZERS

To correct severe phase distortion for PSK data transmission on the
DSCS, an equalizer can be utilized. To correct arbitrary distortion, an adjustable
equalizer is required. A convenient equalizer configuration has the form of a
transversal filter; i.e., a tapped delay line with the tap outputs summed through
adjustable attenuators (the tap weights), as suggested by Figure 5-1.

Mathematically, the transfer function of the equalizer in the frequency
domain is

$$H(j\omega) = \sum_k w_k e^{-j\omega kT} \tag{1}$$

where ω is the frequency offset from center, w_k is the kth tap weight, and T is the tap
spacing. In general, w_k is complex, producing both amplitude and phase control of
the kth tap. Note that (1) is a Fourier series in the variable ω, hence, can represent
an arbitrary transfer function within the range $-\pi/T < \omega < \pi/T$. Thus, both amplitude
and phase distortion can be corrected by the equalizer. Bandwidth restriction is one
special case of amplitude distortion, and is discussed further below.

For digital data transmission by PSK, it can be shown that the transversal
filter (1) in cascade with a receive filter matched to the channel pulse response is the
optimum linear filter in the receiver to combat the intersymbol interference due to
the channel distortion. However, in a practical sense, the receive filter is usually
matched to the transmit pulse; i.e., is an integrate-and-dump detector. Thus, a
transversal filter equalizer can be directly utilized with a typical QPSK/BPSK
modem, which utilizes integrate-and-dump detection. Alternatively, the receiver
can employ a sampling detector following a narrow filter. This would be found in a
modem which incorporates the equalizer as an integral part of the design to
simultaneously correct phase distortion and, like a matched filter, minimize the
effect of receiver noise. (This suggests the terminology "equalizing matched
filter".)
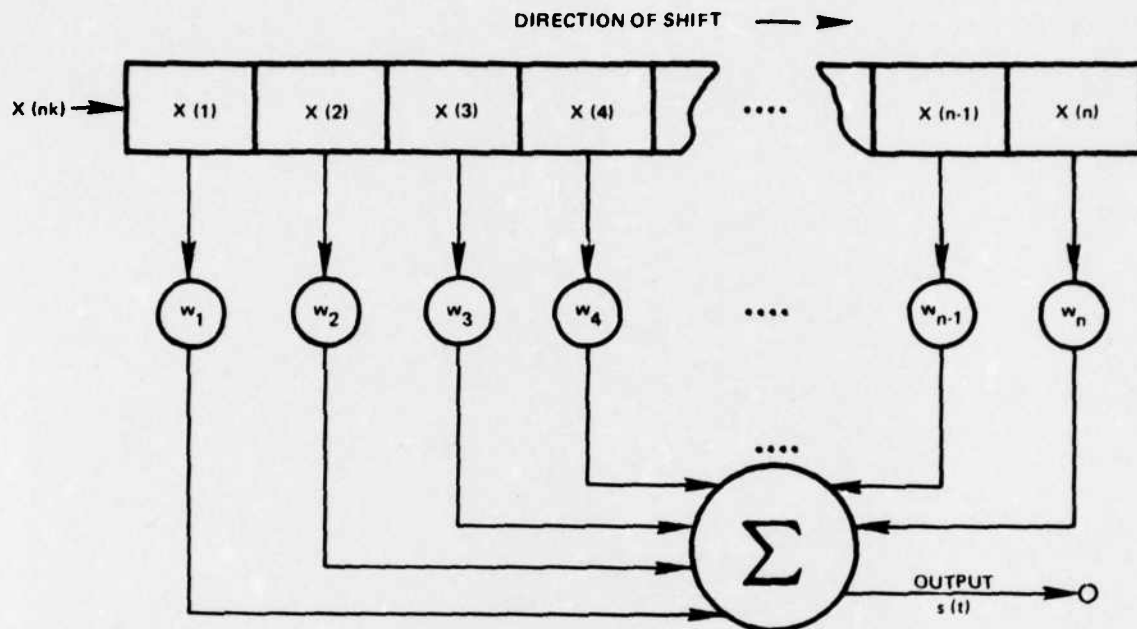
175-99
UNCLASSIFIED

Figure 5-1. Transversal Filter

     If the tap weights can be adjusted during actual data transmission, an adaptive equalizer is achieved. Figure 5-2 shows the concept of automatic adjustment controlled from the error between the actual equalizer output $s(t)$ and the desired output $d(t)$. If the desired output is known a priori, the equalizer is controlled by a "training sequence". If the desired output is recreated from output data decisions, the equalizer control is "decision-directed".

     The former approach applies to a system in which a known training sequence is periodically transmitted solely to equalize the channel. This sequence is known at the receiver and lasts long enough for the equalizer to converge properly. Between such training sequences, the equalizer weights remain fixed at the values established during transmission of the training sequence. If the channel characteristics vary infrequently such a procedure may be adequate for good equalization. However, this method is wasteful since time must be dedicated to the training sequence before useful information can be transmitted over the channel. In addition, changes in channel characteristics occuring between training transmissions can produce unsuspected performance degradations. The major advantage of this technique is that it insures a good equalization of extremely severe channel distortion.
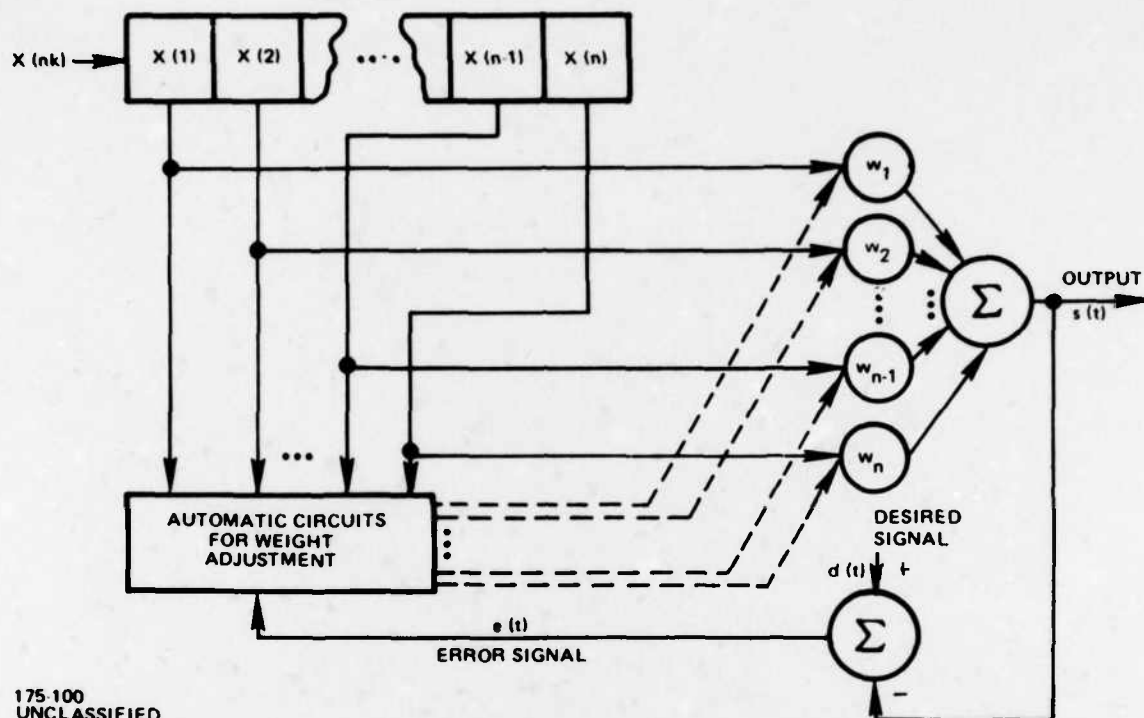
Figure 5-2. Block Diagram of Adaptive Equalizer

The other approach generates the reference signal based on some function of the actual equalizer output. Such a decision-directed approach negates the requirement for periodic transmission of a training sequence since the equalizer "trains" on the actual PSK data signal. This approach has the advantage of permitting continuous adaptation and eliminates the necessity of wasting channel capacity by transmitting training sequence data. In addition, decision-directed techniques can simplify hardware and modem interfaces. These techniques assume that essentially correct decisions on the output data are made to prevent convergence problems. If the channel distortion is severe or the signal-to-noise ratio too low, start-up problems can arise.

Equalizer control based on minimization of a measurable error criterion, such as mean square error, is a practical necessity. Appendix E compares QPSK performance when the channel is equalized to minimize mean square error with the best performance achievable by linear equalization. For the typical example studied, there is essentially no difference.

5.1   FUNCTIONAL CONFIGURATIONS FOR EQUALIZERS

This section describes several functional configurations for an adaptive equalizer to correct phase distortion in the channel. The objective here is to assess practical feasibility, identify interface problems, and consider the problem of deriving control for the adaptation.

5.1.1   BASEBAND EQUALIZER IMPLEMENTATION

In previous equalizer developments for telephone channels, a baseband implementation is utilized. Extended to the satellite channel and QPSK modulation, Figure 5-3 results. Note that coupling between I and Q due to unsymmetrical channel distortion must also be corrected by the equalizer. The configuration in Figure 5-3 requires the equalizer to be connected internally in the modem between the coherent demodulator (phase detectors) and the bit sync and decision circuits. Thus, to equalize an existing modem, a new interface requirement has been created, and the modem must be correspondingly modified to allow the equalizer to be connected internally.

A second disadvantage is that the equalizer has to be implemented with wideband delay lines and summing amplifiers at baseband (or with high speed digital logic). For a 40 MHz satellite channel, the baseband is 20 MHz. There is an implementation problem also associated with performing wideband correlation at baseband to derive the tap weight control.

The primary reason to consider this baseband configuration is its compatibility with adaptive control by the LMS algorithm, described in Appendix C. This algorithm applies to a system forming a linear combination of several signal inputs. Here, these are the delay line tap outputs, which are weighted and summed to provide the equalizer output. The difference between the actual equalizer output and the desired output is the error. The LMS algorithm controls each tap weight proportional to the integral of the cross-correlation between tap error in the equalizer output and the complex conjugate of a tap output. (This control is not shown in Figure 5-3.) In the absence of a training sequence, the error is available on-line in a decision-directed sense by subtracting a reference consisting of hard bit decisions from the unquantized amplitudes prior to making the bit decisions.

Figure 5-3 shows four sets of weights; however, there are actually only two independent sets forming, respectively, the real part and the imaginary part of a single set of complex weights. Nevertheless, it is probably most convenient to
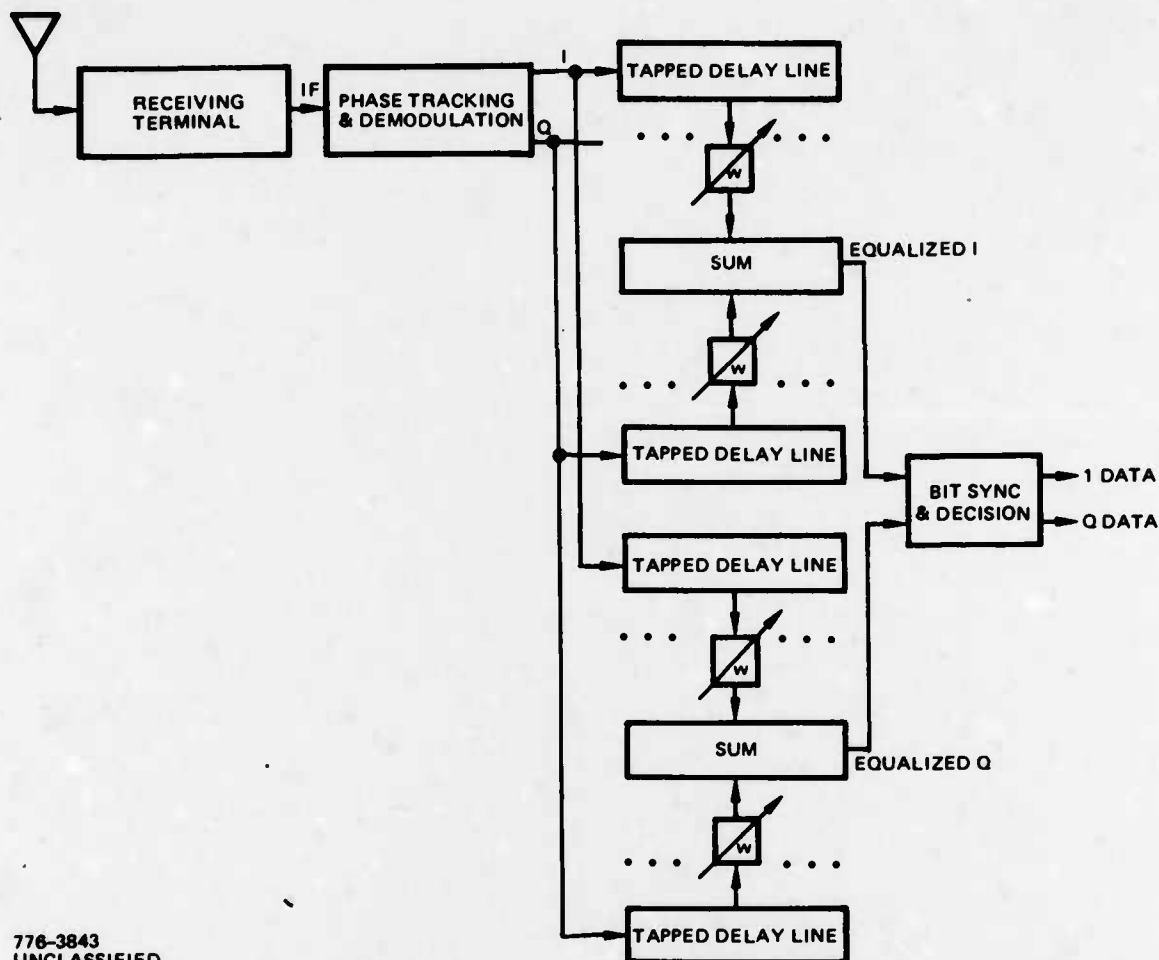
Figure 5-3. Baseband Equalizer Configuration

treat the weights as being all independent and to separately control them, since this compensates for differences between the different delay lines.

In a digital implementation of the baseband tapped delay lines in the form of shift registers, the tap spacing can be a variable set by the modem data clock. The modem I, Q outputs must be quantized to a sufficiently large number of amplitude levels that an accurate equalization is achievable. Consequently, the shift registers are necessarily multilevel. This type of implementation would have the advantage of working at an arbitrary data rate set by the modem, but has not been studied further because of its complexity and high speed requirements.

5.1.2   IF EQUALIZER IMPLEMENTATION

Alternatively, the equalizer can be implemented in the IF between terminal and modem, as shown in Figure 5-4. The interface problem now is minimal. The IF tapped delay line may be an acoustic surface wave device, or even lengths of coaxial cable. The simplicity of an IF implementation is that passive hybrids can be used for combining the weighted tap outputs. As seen in Figure 5-4, a tap output is split into two isolated outputs which are respectively weighted by the real part and the imaginary part of the complex weight. The weighted real parts from all taps are then combined through a cascade of hybrids; similarly for the imaginary parts. Finally, the combined real parts and the combined imaginary parts are combined through a quadrature hybrid, which imposes a 90° phase shift on one of the inputs.

776-3844
UNCLASSIFIED
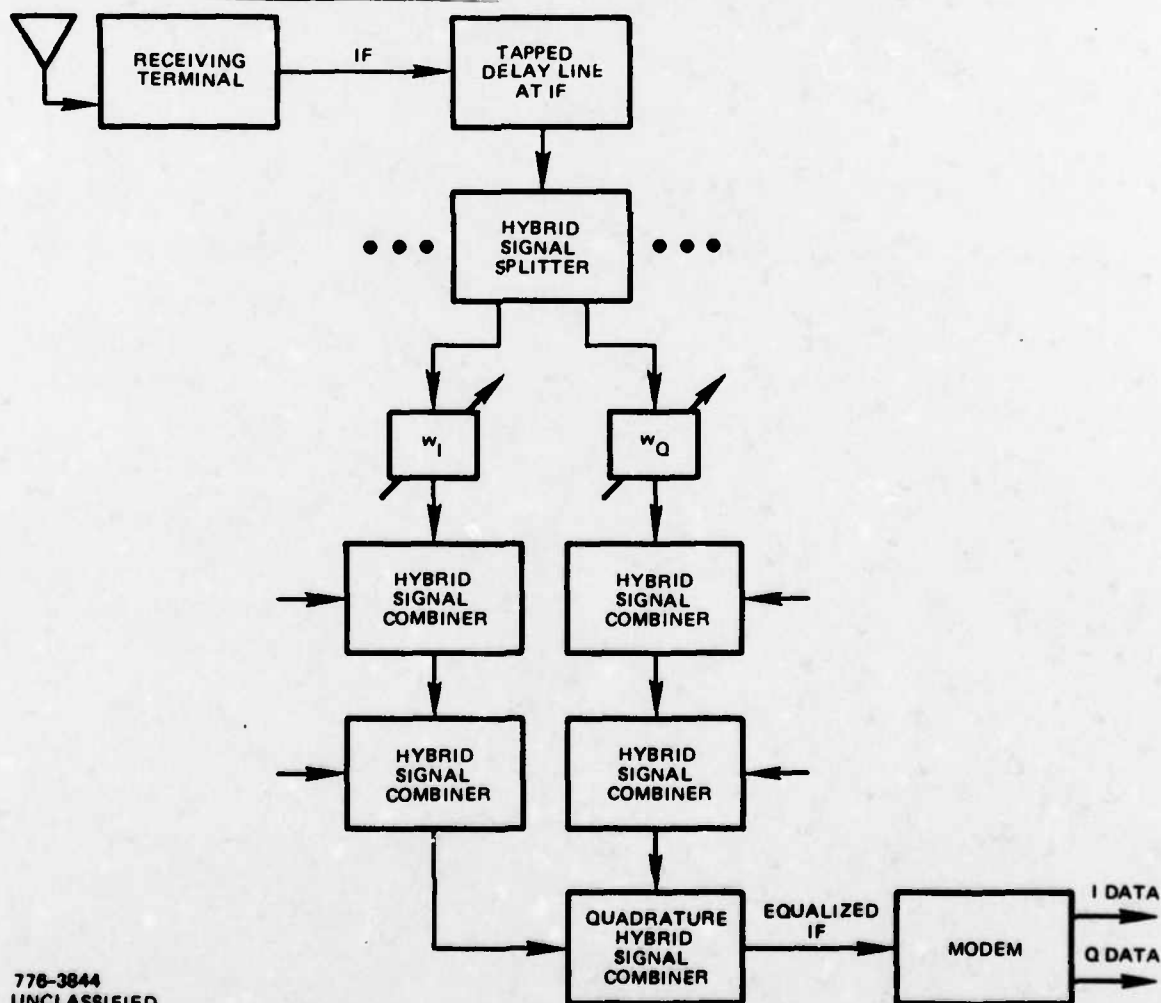
Figure 5-4. IF Equalizer Configuration

Since the adjustable weights are easily implemented by PIN diode attenuators, the IF equalizer configuration appears quite economical. It is basically identical with typical implementation of an adaptive antenna array, hence incorporates familiar and well-tested hardware.

The main disadvantage of the IF equalizer implementation (in contrast to antenna null-steering) is noncompatibility with the LMS algorithm concept.

The LMS algorithm is of the steepest descent type and acts to reduce the mean square error between the equalizer output and the reference. The gradient of the error is proportional to the cross-correlation between the instantaneous error and the complex conjugate of a tap output, and this is the basis for the usual equalizer control. But, implicitly, this control concept assumes no delay between the individual tap outputs and the final equalizer summed output. Furthermore, a reference is needed to compute the error at the summing point which yields the equalizer output.

A distinctly different control concept is now proposed which is more generally applicable, in the sense that the error* can be created and measured at an arbitrary point in the system, not necessarily at the summing point. This control concept "dithers" a tap weight, alternately increasing and decreasing the nominal value of the weight by a small increment. The error is measured for each of the two weight values, and the difference taken. This difference is approximately proportional to the gradient of the error with respect to the nominal value of the tap weight. The tap weights can be sequentially controlled in this manner (either manually or automatically) so as to drive the error to a minimum.** Note that cross-correlation between the error and the tap output is not utilized with this control concept; instead, the gradient is effectively measured by perturbing the tap weight value.

---

*While mean square error is mathematically convenient and is assumed in deriving the LMS algorithm, other error quantities can be utilized for practical convenience.

**Another possibility is to dither many tap weights simultaneously with orthogonal dither functions, such as different frequency dithers.

For application to control of an IF equalizer, the error (in a decision-directed sense) can be measured at the modem output by comparing the unquantized I and Q output samples with hard decisions on these samples. However, this presumes an interface with the modem such that the unquantized modem outputs can be made available externally. Figure 5-5 describes the dither configuration, where the IF equalizer block is that shown in detail in Figure 5-4.

The soft decision outputs available from a typical QPSK/BPSK modem designed to interface with a Viterbi decoder are quantized to 3 bits, in accordance with Figure 5-6 (which is a duplicate of Figure 3-14). Computer simulations of the PSK channel show that Viterbi decoder performance for 3-bit quantized soft decisions is only slightly worse than for unquantized soft decisions, and a 2 dB improvement is achieved with respect to hard decisions. (See Section 4.3.1.)

In the absence of error correction coding, the quantized soft decisions yield information on modem error rate performance. Specifically, "alpha flunk" is defined as the fraction of bit decisions which are less than 0.5 amplitude. Thus, alpha flunk can be obtained simply by counting the second bit of the 3-bit quantized output. Figure 5-7 shows the theoretical relation between alpha flunk and uncoded

776-3845
UNCLASSIFIED
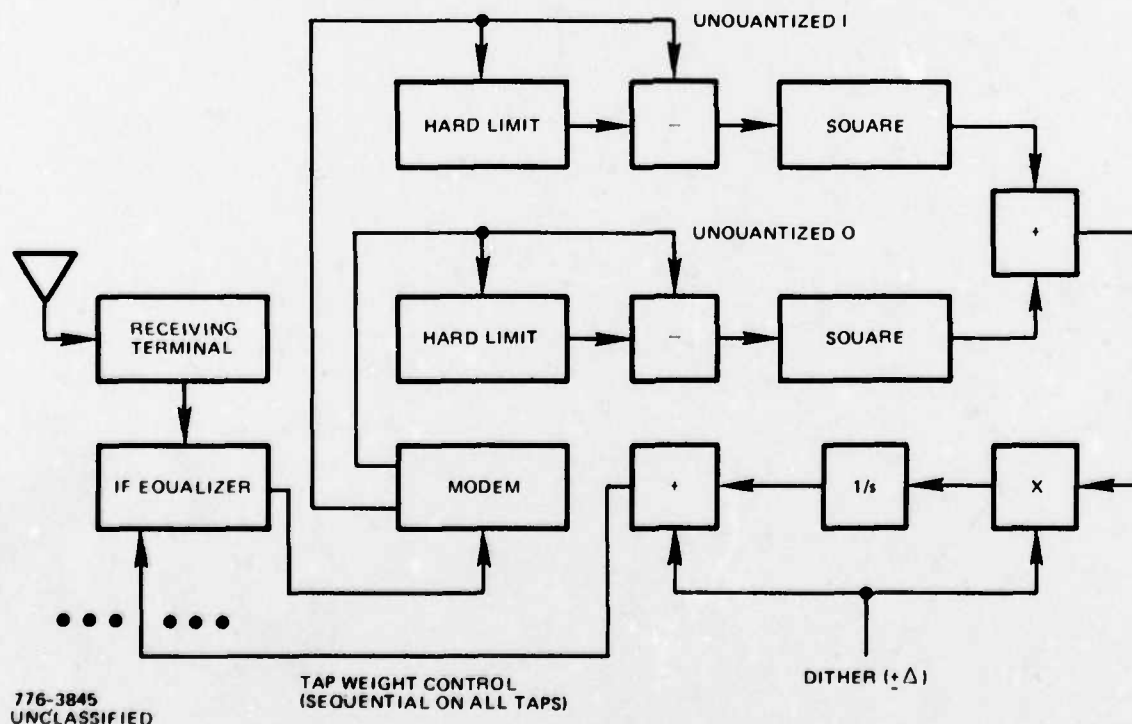
Figure 5-5. Sequential Dither Control of Tap Weights
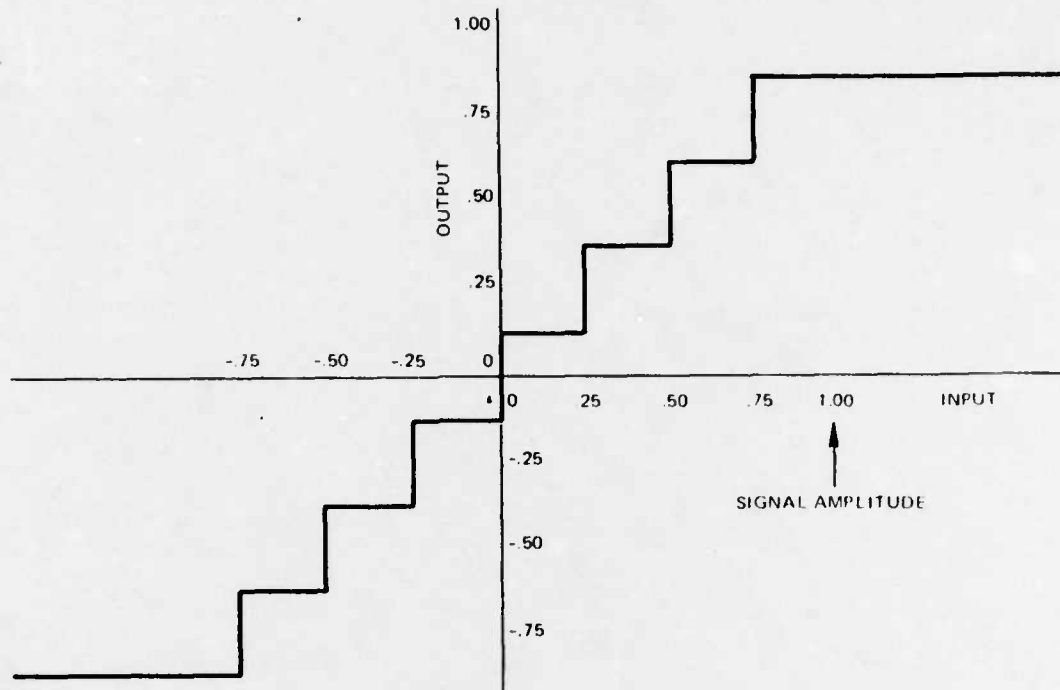
776-3583
UNCLASSIFIED

Figure 5-6. 3-Bit Quantization for "Alpha Flunk" Information
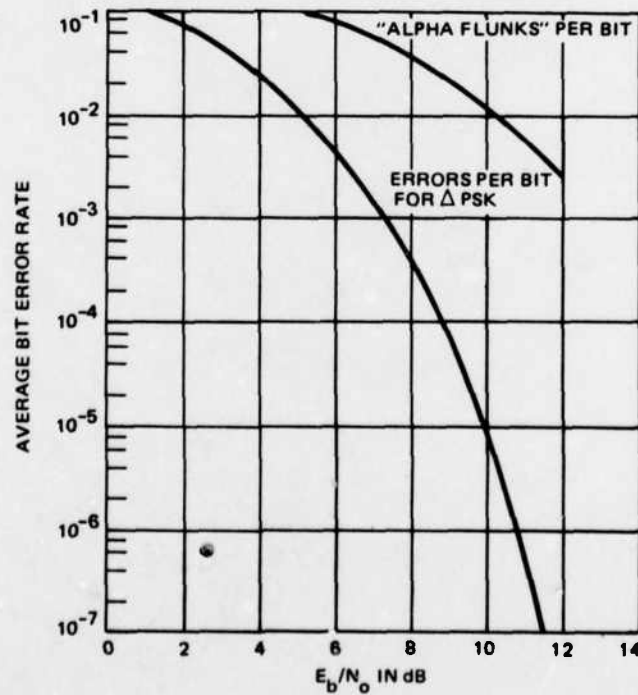
776-3847
UNCLASSIFIED

Figure 5-7. Relationship of "Alpha Flunks" to BER and E_b/N_o

error rate of PSK. Alpha flunk is a particularly convenient error criterion which may be applied to control an IF equalizer.

An alternative technique for reconstructing a reference to control an IF equalizer is based on hard limiting, as shown in Figure 5-8. That is, the equalized IF output is hard limited, and the hard limiter output is subtracted out to give the error. This error is cross-correlated with the complex conjugate of each tap output to control the tap weights via the standard LMS algorithm. Note that a special interface with the modem is not required with this technique.

This scheme is predicated on the fact that the signal is constant envelope (BPSK or QPSK) until the channel introduces the phase distortion. After the distortion, the signal displays envelope fluctuations due to the intersymbol interference. The envelope fluctuations are minimized by the action of the LMS algorithm controlling the equalizer tap weights.

An obvious question is whether some channel filter or dispersion can be postulated such that BPSK or QPSK is significantly degraded while remaining approximately constant envelope so that the equalizer converges to an incorrect transfer function; hence, this technique must be studied further with realistic simulations.
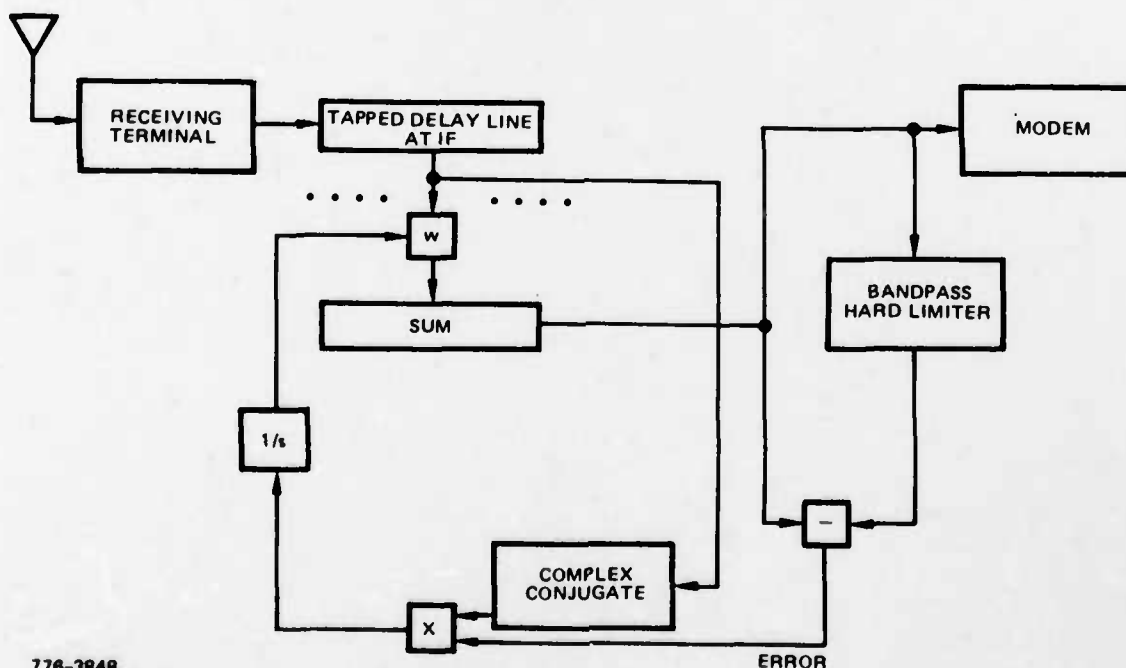
776-3848
UNCLASSIFIED

Figure 5-8. Hard Limiter to Construct Reference at IF Without Using Modem

5.1.3 TRANSMISSION OF PSEUDORANDOM PILOT SIGNAL

A completely different approach to on-line equalization eliminates need for decision-directed control by transmitting a known pseudorandom pilot signal at a low level (say 20 dB down) so that it does not interfere significantly with the QPSK data signal. In the receive IF, phase lock and delay lock to the pilot signal is achieved by a rudimentary PN spread spectrum receiver, consisting only of tracking loops without need for data demodulation. Now, the reference is the replica pseudorandom code (biphase or quadriphase) modulated on the reconstituted carrier, and the error is taken between the equalizer output at IF and the reference. By making the PN periodic, it is possible to integrate over many repetitions to eliminate interference caused by the data signal, and also this makes the initial synchronization easier to find.

This technique has not been studied further because it requires transmitter modification.

5.1.4 DECISION-DIRECTED EQUALIZING MATCHED FILTER

In a prior study of high rate QPSK[6], feasibility was demonstrated of using a tapped delay line as an adjustable transversal filter to equalize channel distortion and simultaneously form a matched filter for minimizing the effect of additive noise in the receiver. The basic idea is to derive control from the error between the equalizer output and the reference ideal output from a matched filter. As discussed in Appendix C, the LMS algorithm control computes the cross-correlations between this error and the various tap outputs, and adjusts each tap weight to be proportional to the integral of the corresponding cross-correlation. Note that the reference ideal matched filter output consists of a ramp* wherever a bit transition occurs.

For a training sequence of known bits, there is no conceptual difficulty in constructing the reference ideal output; the process is one of linear interpolation between successive known bit values, and the relative timing of the training sequence can be set arbitrarily**. However, a problem arises in connection with decision-

---

*The ramp is the result of the integrate-and-dump matched filter process operating on the instantaneous polarity reversal in the bit stream at a bit transition.

**The relative timing would typically be adjusted to minimize edge effects from a finite length equalizer; of course, the training sequence must be generated at the same rate as the received bit stream, necessitating a clock tracking loop.

directed control of the equalizer; namely, that when the reference ideal output is reconstituted from the modem's bit decisions, it is inherently delayed by one bit duration. Thus, the equalizer output must be correspondingly delayed by a bit duration to obtain the error, and the tap outputs must be similarly delayed for computing the cross-correlations.

Since an application of the equalizing matched filter is to handle variable data rates achieving near-optimum performance without changing integrate-and-dump filters,* having to implement variable delays dependent on the actual data rate is a significant disadvantage.

The following describes a decision-directed equalizing matched filter which does not require delays matched to the bit duration in order to implement the LMS algorithm control. Furthermore, as shown in Section 7.1.2, degradation arising from a narrow channel or receiver bandwidth is reduced by the equalizing action. This decision-directed equalizing matched filter appears to give a desirable implementation for a high rate QPSK/TDMA modem.

5.1.4.1   System Concept

The system concept presented here is predicated upon controlling the tap weights of the equalizer so as to minimize the mean square error in the bit decision samples. Then, ignoring intersymbol interference for the moment, the equalizer's transfer function must approach an approximation to a filter matched to the bit duration, since this minimizes the error caused by the additive noise.** Now, with channel distortion also present, the equalizer will effect a compromise between noise and intersymbol interference on the bit decision samples, in the process of minimizing the total mean square error.

We see that the system concept here is quite different from that described in reference 6. There, the reference waveform from which the error is formed is deliberately designed to force the equalizer towards approximating a filter matched to the bit duration. Here, this action is a natural consequence of forming the error on the bit decision samples, with sampling at the bit rate, in combination with the mean square error minimization property of the LMS algorithm.

---

*In fact, the results in reference 6 are predicated on having an integrate-and-dump filter matched to the tap spacing, which is fixed independent of actual data rate.

**A theoretical proof of this assertion would be nice to have; however, it seems intuitively acceptable, and is supported by the results of computer simulations given below.

Figure 5-9 sketches the conceptual block diagram of the equalizing matched filter realized by a tapped delay line at IF*, with adaptive control of the tap weights by the LMS algorithm. The control of each complex weight is proportional to the integral of the cross-correlation between the complex conjugate of the tap output and the decision-directed complex error on the bit decision samples. For QPSK, separate hard decisions are made on the real and imaginary parts of the equalizer sum output (i.e., in-phase and quadrature components); then, the error with respect to these decisions is taken.

It is assumed in Figure 5-9 that the input has already been translated to the correct IF so that the equalizer control loops need only track slow changes in phase. The complex weight on a tap output gives control of both amplitude and phase into the sum. A carrier tracking loop, such as a Costas loop for BPSK or a quadrupling-type loop for QPSK, can be provided for the purpose of translating to the correct frequency, but the equalizer does the exact phase tracking.

The data clock must, of course, be recovered from the signal. Figure 5-9 indicates a delay-lock loop which tracks the bit transitions in the data. As usual for a clock tracking loop, the bit transitions have to be polarity corrected (i.e., positive-going segregated from negative-going transitions). Note that the output of the equalizing matched filter can be sampled to yield both the bit decision samples and the bit transition samples, the latter occuring at a time displacement of a half bit.

5.1.4.2 Heuristic Design Considerations

The receive bandpass filter is an important system element which is needed to optimally process white noise on the input signal. Ideally, the filter should have a transfer function corresponding to integrate-and-dump matched to the tap spacing of the tapped delay line. Practically, the receive bandpass filter will be sharp cutoff, creating intersymbol interference (which will be corrected by the equalizer). By Nyquist sampling theory, the bandpass should not exceed the reciprocal of the tap spacing; otherwise, we expect a performance degradation due to excess noise power.

---

*Alternatively, the equalizer can be implemented as four tapped delay lines at baseband. [7, 8, 9]
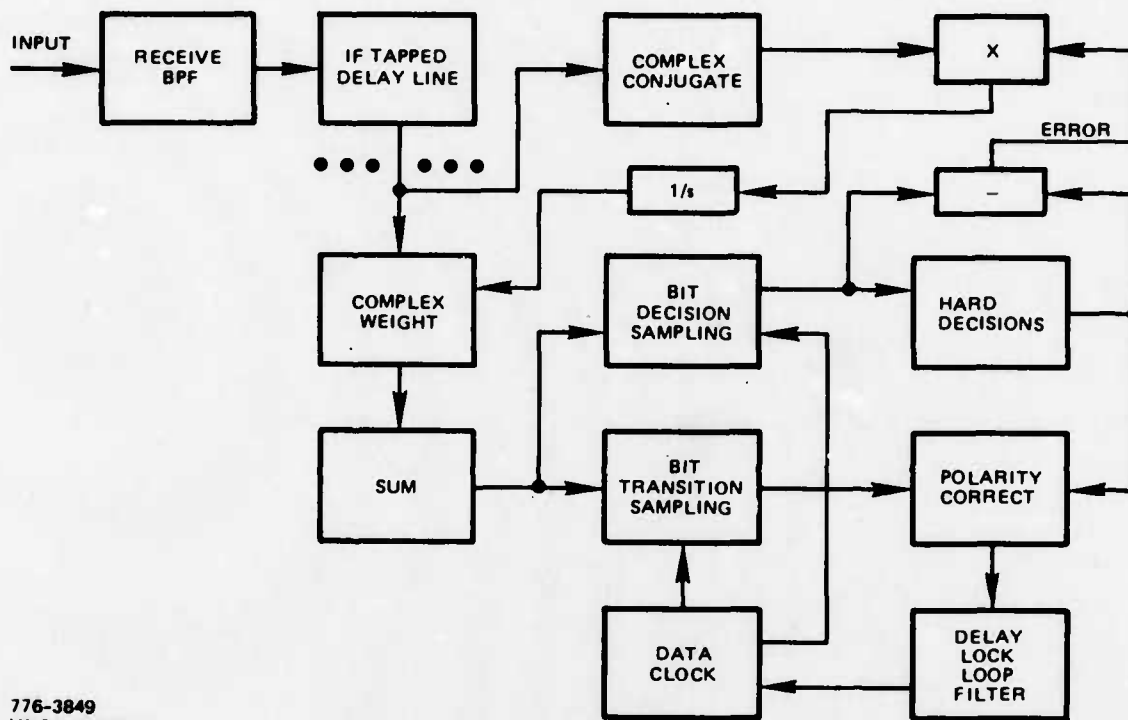
776-3849
UNCLASSIFIED

Figure 5-9. Concept of Equalizing Matched Filter With LMS Algorithm Control
on Bit Decision Samples

For an unequalized receiver with a data detector matched to the bit
duration, the bandpass should at least equal twice the bit rate* to avoid significant
performance degradation due to intersymbol interference[10]. Using this receive
bandwidth with the equalizing matched filter, the tap spacing should not exceed half
the bit duration. However, since the equalizer reduces intersymbol interference
while simultaneously creating a matched filter, it is likely that a narrower bandpass
can be employed successfully. Setting the transmit and/or receive bandpass equal to
the bit rate and the equalizer tap spacing equal to the bit duration should therefore be
studied.

It is interesting to make the observation that the receive bandpass filter
reduces noise as well as signal outside the cutoff; hence, the action of the equalizer
to restore spectrum components attenuated by the receive filter does not imply a
significant signal-to-noise ratio penalty.**

*With QPSK, the terminology bit rate and bit duration, as used here, refers to
either the in-phase or the quadrature bit stream.

**If white noise is added after a filter having a spectrum null, performance with
transversal filter equalization tends to be severely degraded.

5-14

5.1.4.3   Theoretical Analysis

A theoretical basis for the equalizing matched filter is now presented showing that the resulting performance in a restricted bandwidth should be close to ideal PSK (either BPSK or QPSK), even for a bandpass as narrow as the transmitted bit rate. Figure 5-10 portrays the system under consideration, without channel distortion except that due to the transmit filter. The equalizer in the receiver acts to reduce the effect of intersymbol interference without introducing excessive noise. The transmitted power is normalized at the output of the transmit filter.
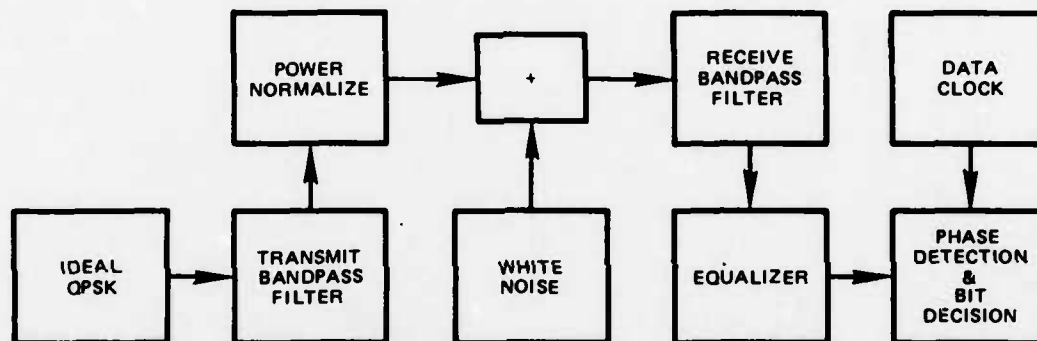
776-3850
UNCLASSIFIED

Figure 5-10. Model of PSK Bandlimited Channel

We now observe that if the transmit and receive bandpass filters are ideal sharp cutoff with bandpass equal to the transmitted bit rate, and if the input excitation to the transmit filter consists of ideal phase-modulated impulses, ideal QPSK performance can be realized without equalization. The bit decisions are sampled setting the sampling clock phase where intersymbol interference from adjacent impulses is zero. The band-limited channel impulse response (appearing as modulation on the carrier) is

$$h(t) = \frac{\sin(\pi t/T_b)}{\pi t/T_b} \tag{2}$$

for unit power normalization. *

---

*For power S, the sampled amplitude at $t = 0$ is $\sqrt{S}$. The low pass equivalent of the bandpass has cutoff at $1/2T_b$; hence, the noise power is $N_o/2T_b$. The power ratio is $2ST_b/N_o = 2E/N_o$, the ideal value for a matched filter.

The role of the equalizer is now evident; namely, that it must compensate for transmitter excitation by finite-width pulses instead of true impulses. As a lower bound on achievable performance, assume that the equalizer restores the flat spectrum corresponding to excitation by true impulses.* Note that an arbitrary phase distortion can, at least in principle, be completely removed by the equalizer; hence, the equalizer is forcing the intersymbol interference to be zero.

The power spectrum of a PSK signal with bit duration T_b is

$$S(f) = T_b \left[\frac{\sin(\pi f T_b)}{\pi f T_b} \right]^2 \tag{3}$$

which has unit power over an infinite bandwidth. After transmit filtering to the passband $-1/2T_b < f < 1/2T_b$, equal to the bit rate, the power remaining is 0.774, which is restored to unity by the normalization process in the transmitter. Since the desired flat power spectrum over the passband is T_b, corresponding to a total power of unity, the equalizer is required to have a power response

$$\left| Y(f) \right|^2 = 0.774 \left| \frac{\pi f T_b}{\sin(\pi f T_b)} \right|^2 ; \quad -1/2T_b < f < 1/2T_b \tag{4}$$

After equalization, (2) gives the output time response to a finite-width pulse input, and there is no intersymbol interference on a bit decision sample at $t = 0$.

The equalizer causes an increase in the noise power on the bit decision sample. The increase is obtained by integrating (4) over the passband. A numerical integration yields a noise power increase by the factor 1.084. This means a degradation of only 0.35 dB from ideal PSK system performance.**

_____

*This is a lower bound because it does not necessarily yield the best compromise between intersymbol interference and noise effects. If the bandwidth exceeds the bit rate, the bound is not a good approximation to achievable performance, since (2) is not the optimum response in this case.

**It is worth pointing out once more that E_b/N_0 is defined with respect to the power output of the transmit filter. Reduction in power due to transmit filtering is not, in itself, a degradation according to this definition.

The above analysis shows that a bandwidth equal to the transmitted bit rate suffices for PSK, and with an equalizer to reduce intersymbol interference in the receiver, nearly ideal PSK performance is achieved. The analysis suggests that the transmit bandwidth should not exceed the receive bandwidth, at least under the assumption of sharp cutoff filters; otherwise, the power transmitted in the spectral components outside the receive passband is wasted. However, with practical filters, this conclusion is not necessarily valid.

With a wider bandwidth, the above analysis can be generalized to obtain an explicit solution for the optimum equalizer which either forces intersymbol interference to zero and minimizes the noise error, or minimizes the total mean square error[11].

5.2   SIMULATION OF DECISION-DIRECTED EQUALIZATION

Control of the complex tap weights of an IF equalizer by means of the LMS algorithm (see Appendix C) is straightforward when the data sequence is known (i.e., use of a training sequence) and the signal is noise-free. The weighted linear combination of the tap outputs may be compared with the reference signal, which is an ideal PSK waveform modulated with the known data, to create the error waveform. Each tap weight then is adjusted proportional to the cross-correlation between this error waveform and the complex conjugate of the tap output, and the process is iterated until convergence is obtained. The equalized signal may then be demodulated and the error probability computed. * It is assumed that the receiver tracks carrier phase and the data clock prior to the equalization process, so that the idealized summing structure needed for the LMS algorithm is actually valid.

A study of the benefits of equalization for this idealized structure assuming a training sequence for the reference is presented in Section 6. In addition, the technique mentioned in Section 5.1.2 of hard limiting the output of the IF equalizer to create a reference for the LMS algorithm is also studied in Section 6, again for noise-free operation.

This section examines equalizer control for decision-directed operation during on-line data transmission. In particular, the effect of receiver noise on the adaptation process is included in the various computer simulations. A significant

---

*Note, noise is not actually added for this overall process. The noise power additive to the demodulator output is computed from the transfer function of the equalizer in cascade with the receiver.

difference between decision-directed operation and training sequence operation is that the former utilizes the entire data bit to make the decision; hence, there is an inherent lag which prevents a direct subtraction of the equalized signal and the reference created from the bit decisions. Consequently, the error for decision-directed control is defined for the bit decision samples (inphase and quadrature) obtained from the demodulator rather than for the entire waveform.

Figure 5-11 sketches the concept of decision-directed equalizer control. Note that the error is defined with respect to hard decisions made on the polarities of the samples.

A basic question raised by the configuration of Figure 5-11 is the effect of modem tracking within the feedback control loop on the equalizer tap weights. The capability within the modem to respond to changes in carrier phase and bit timing must not destroy the measurement of changes in error needed to control the equalizer.
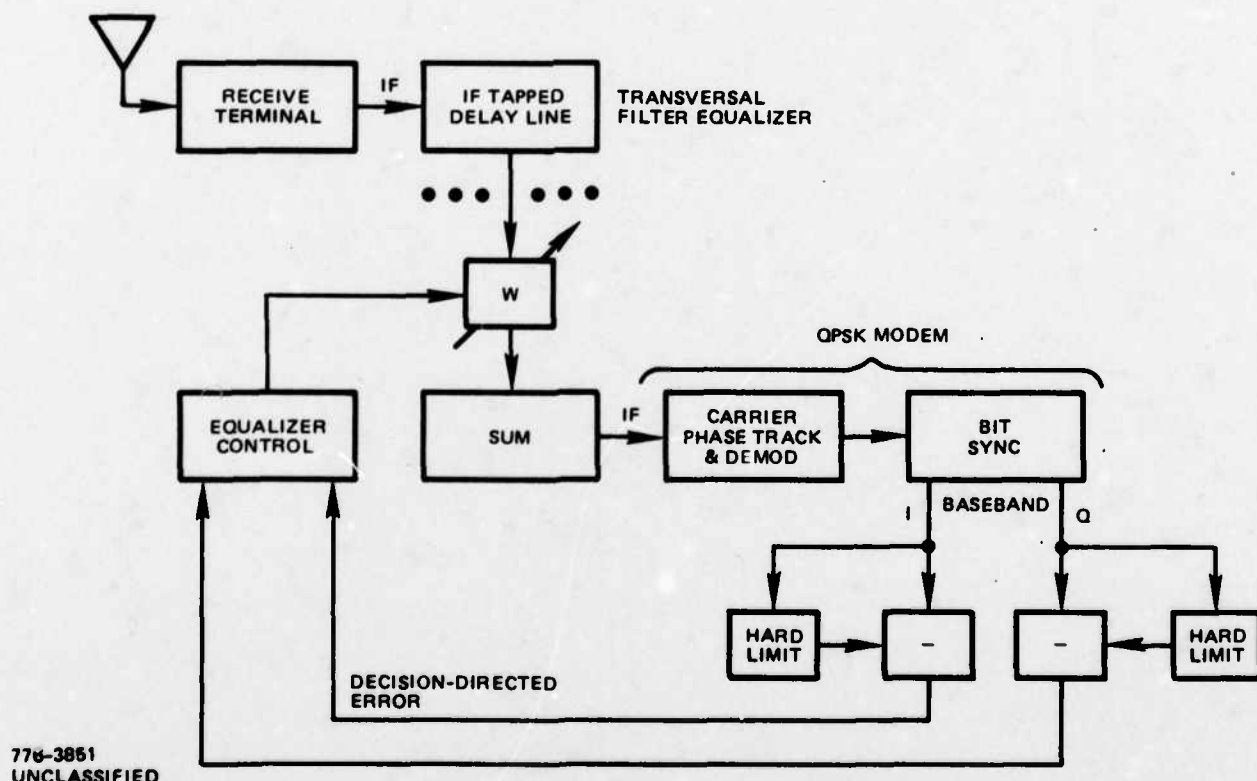
Figure 5-11. IF Equalizer Controlled by Error in Modem Output

5.2.1   VALIDATION OF EQUALIZER CONTROL CONCEPT

A computer simulation was performed to assess the basic validity of utilizing an IF transversal filter equalizer controlled by the error in the modem output samples. The general control concept is to measure the gradient of the error in the output with respect to the tap weights, and adjust the weights to reduce the error toward a minimum.

In this simulation, the modem incorporates an integrate-and-dump filter, whose output is sampled synchronously with the transitions in the bit stream. A generic model of the bit synchronizer samples the integrate-and-dump filter at the midpoints between the sampling times for extracting the data, and corrects each sample for the polarity of the bit transition. This yields a clock tracking error voltage to maintain lock to the bit timing. Also, of course, the modem tracks carrier phase by a quadrupling-type loop for QPSK.

Compared with the standard theory of adaptive combining leading to the LMS algorithm, as described in Appendix C, the present case is characterized by the existence of a filter (the integrate-and-dump detector) located between the summing point and the point at which the error is observed. Consequently, we need to extend the theory to handle this new situation. Let y denote the channel output and z the output of the transversal filter equalizer, so that

$$z_k = \sum_{m=1}^{M} w_m \, y_{m+k} \tag{5}$$

where w denotes the set of equalizer tap weights to be adjusted. The observable quantity is the modem output u, the result of z passing through the integrate-and-dump filter. That is,

$$u_k = \sum_{i=1}^{I} z_i \, g_{k-i} \tag{6}$$

where g is the impulse response of the filter. If the desired output is d, the error at the kth sampling time is

$$\epsilon_k = d_k - u_k \tag{7}$$

and the gradient of $\left| \epsilon_k \right|^2$ with respect to the tap weights is needed.

In an actual equalizer implementation, the tap weights can be "dithered" to measure the gradient of the error, so as to adjust the weights to decrease the error. For a computer simulation, however, it is more convenient to express the derivatives directly, and employ the LMS algorithm to adapt the tap weights. Differentiating the square error $|\epsilon_k|^2$ with respect to the real and imaginary parts of a tap weight, we obtain

$$\frac{\partial |\epsilon_k|^2}{\partial \text{Re}\{w_m\}} + j \frac{\partial |\epsilon_k|^2}{\partial \text{Im}\{w_m\}} = \left[\sum_{i=1}^{I} y_{m+i}^* g_{k-i}^* \right] \epsilon_k \tag{8}$$

which cross-correlates the error in the modem output with the complex conjugate of the tap output passed through the same integrate-and-dump filter. In an actual equalizer implementation, the tap outputs are, of course, not filtered; hence, the LMS algorithm cannot be applied directly. However, in a computer simulation, (8) can be computed simply by reversing* the order of the modem filtering and the equalization; in which case, the standard LMS algorithm is then applicable to adjust the tap weights. A dither approach to control the tap weights in an actual equalizer implementation should behave similarly, since it has the same criterion of minimizing the error in the modem output samples.

The computer simulation generates 1024 samples of QPSK, with 32 samples per QPSK symbol. Thus, there are 32 QPSK symbols in one period of the simulation. This signal is passed through the presumed channel distortion. Then, for an arbitrary phase of the bit synchronizer, the channel output is passed through an integrate-and-dump filter matched to the 32 sample duration. The output of the integrate-and-dump filter then passes through the transversal filter equalizer. The equalizer output is sampled at the 32 bit times corresponding to the arbitrary clock phase of the bit synchronizer, and a QPSK decision is made on each sample. The decision-directed error in each sample (i.e., bit amplitude subtracted from a hard bit decision) is used to adjust the tap weights via the LMS algorithm. Note, since the equalizer is at IF and a complex signal representation is employed, the adjustment of the complex tap weights includes the effect of modem phase tracking.**

---

*With cascaded linear filters, the order of filtering is immaterial as far as the final output is concerned.

**If the equalizer adapts more rapidly than the rate at which the modem can track changes in carrier phase, the above statement is strictly true, and the equalizer compensates for phase tracking error in the modem. More likely, the equalizer will adapt more slowly; however, presumably the modem is normally able to track with a small phase error.

Midway between the points at which the bit amplitudes are sampled, the equalizer output is sampled to obtain the clock tracking error, as described above. The clock tracking error is averaged over the period of the signal (32 QPSK symbols). Figure 5-12 gives a plot of the clock tracking error as a function of the sampling time for a particular channel distortion, both prior to equalization and after the IF equalizer has adapted. It is found that the mean square error in the data samples was approximately minimized at the sampling time for zero tracking error; however, good equalization still can be achieved for a relatively large tracking error. After equalization for the presumed channel distortion (at a high E_b/N_o) the rms error in the bit samples is approximately 2 percent for the optimum sampling time. Before equalization, the rms error is 7 percent.

The significance of the curve in Figure 5-12 of clock tracking error after equalization is that the clock tracking loop in the modem can be allowed to have a bandwidth wider than the adaptation rate of the equalizer. Although Figure 5-12 shows some reduction in slope, compared to the unequalized case, this should not impair operation of the modem's clock tracking loop. Furthermore, the equalizer will tend to compensate for clock error in the modem. To show this, Figure 5-13 plots the

T_s = QPSK SYMBOL DURATION
CHANNEL DISTORTION =
$-.3 + e^{-j\omega T_s/4} + .3e^{-j\omega T_s/2}$
EQUALIZER HAS 16 TAPS
WITH SPACING $T_s/2$
A CENTER TAP IS INITIALLY NONZERO

UNEQUALIZED
EQUALIZED
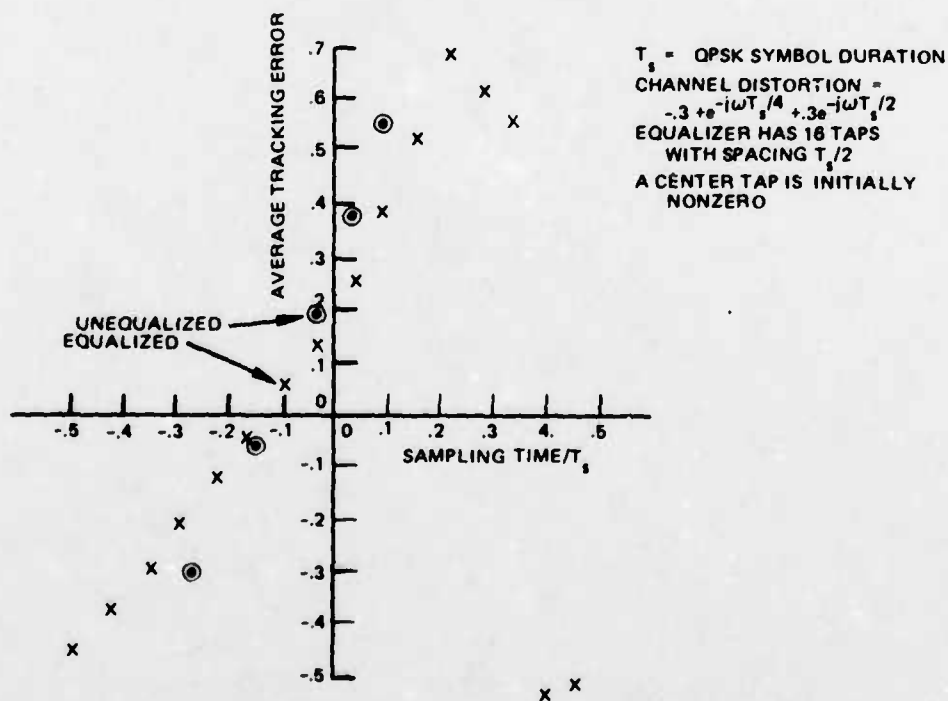
1176-6033
UNCLASSIFIED

Figure 5-12. Clock Tracking Error After Decision-Directed Equalization on Data Samples. High E_b/N_o

5-21
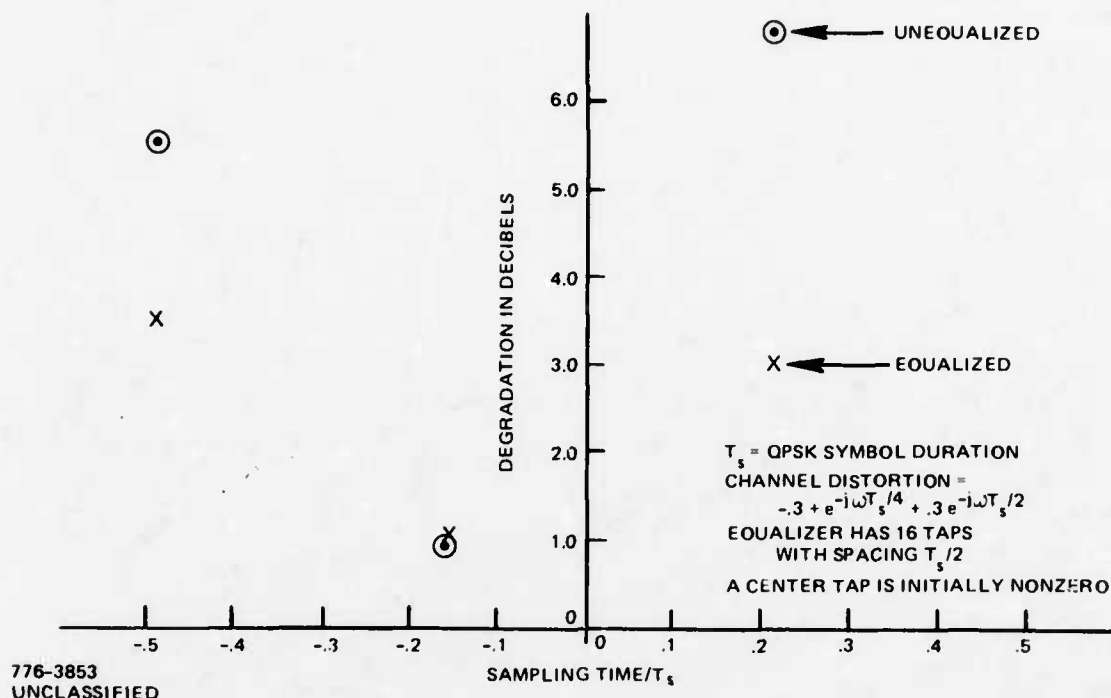
Figure 5-13. Degradation From Ideal QPSK as Function of Sampling Time, at $E_b/N_o = 5$ dB

degradation as a function of sampling time, at $E_b/N_o = 5$ dB. The degradation is defined relative to the ideal value of E_b/N_o for the observed QPSK error probability, where white Gaussian noise is presumed. (The effect of the equalizer's frequency response on noise power at the output of the integrate-and-dump filter is taken into account by a numerical integration in the frequency domain.) Note how equalization reduces the sensitivity to an error in clock phase. (At the optimum sampling time, the improvement due to equalization is small because the assumed phase distortion is minor.)

The sampling time for zero tracking error in Figure 5-12 is in good agreement with the lag in the channel. The phase of the channel distortion presumed in Figure 5-12 is

$$\theta(\omega) = -\omega T_s/4 - \arctan[.6 \sin(\omega T_s/4)] \qquad (9)$$

and the time shift is ....

$$\frac{d\theta}{d\omega}\bigg]_{\omega=0} = -0.4 \, T_s \qquad (10)$$

The shift in the transversal filter is $3.5 \, T_s$, so that the total shift is $3.1 \, T_s$. The tracking error is, of course, periodic in T_s; hence, $3.1 \, T_s$ is equivalent to a sampling time of $-0.1 \, T_s$, which closely agrees with the null point observed in Figure 5-12

Figure 5-14 is similar to Figure 5-13 except for a greater phase distortion within the significant bandwidth of the signal. Again, the minimum degradation after equalization is found to occur near the null point for clock tracking.
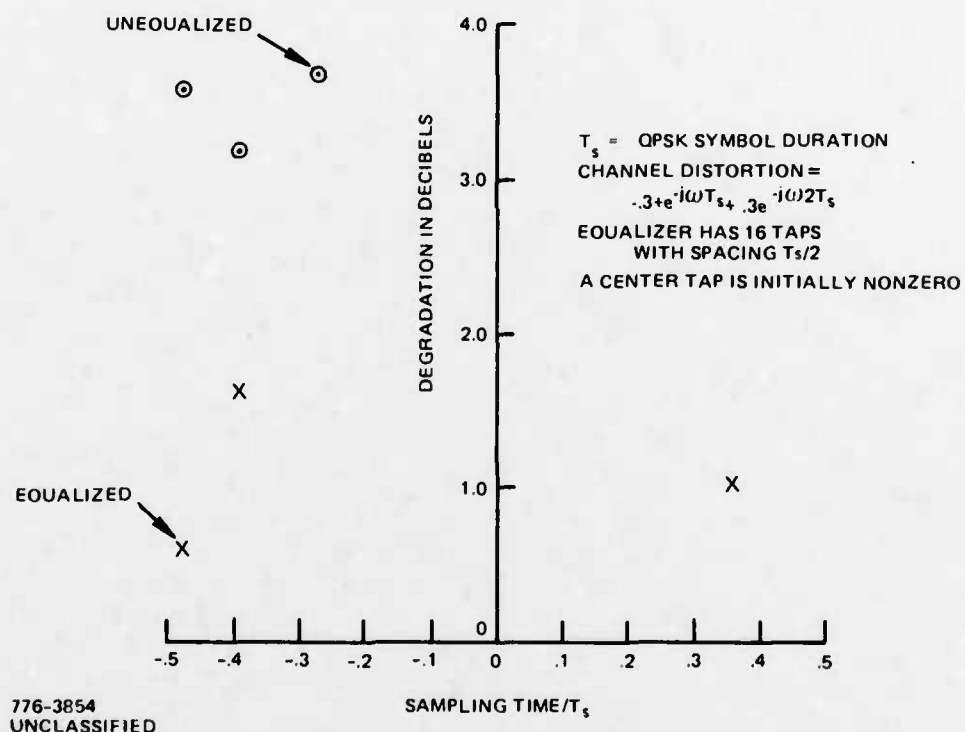
Figure 5-14. Degradation from Ideal QPSK as Function of Sampling Time, at $E_b/N_o = 5$ dB

It is concluded that the concept of adapting an IF transversal filter to equalize the channel using the decision-directed error in the modem output data samples is valid. Bit synchronization information remains available in the equalized signal, so that the bit synchronizer in the modem will still function properly despite the feedback control loop. Moreover, the equalizer will tend to compensate for clock tracking error in the modem.

5.2.2 SIMULATION OF DITHER CONTROL OF IF EQUALIZER FROM MODEM OUTPUT

On-line equalization of a QPSK communication channel by means of a tapped delay line at IF with control of the tap weights based on the dither technique is studied further in the following. Figure 5-5 illustrates the basic principle of decision-directed dither control of the equalizer tap weights, using the sampled baseband output of the modem. By alternately increasing and decreasing a tap weight and observing the effect on the error in the output samples, the dither control attempts to minimize the mean square error in the output samples, where the error is taken with respect to hard decisions on the samples. Figure 5-5 presumes the unquantized samples are available from the modem on an existing or easily created interface; however, this may not be the case for QPSK modems in the DSCS. Therefore, it is also of interest to investigate the possibility of equalizer control from the 3-bit quantized output of the modem, designed to interface with a soft-decision Viterbi decoder.

5.2.2.1 Simulation Model of Dither Control

As discussed above in Section 5.2.1, the integrate-and-dump detector of the modem can precede the IF equalizer in the computer simulation, since the order of filtering is immaterial in a linear system. In the simulation, this enables control of the tap weights by the LMS algorithm; i.e., changing a tap weight proportional to the cross-correlation of the error with the complex conjugate of the tap output. The assumption here is that the LMS algorithm will converge to the same set of equalizer weights as the dither technique, but with the advantage of requiring less computation time for the simulation.

Figure 5-15 is a block diagram of the QPSK computer simulation model where control of the equalizer tap weights w_m is via the LMS algorithm. Bandpass signals are represented by complex samples. Since the equalizer has arbitrary complex weights, it can perform the phase tracking function normally done by a quadrupling-type loop in the modem. In other words, adjusting the equalizer for error minimization with respect to hard decisions on the real and imaginary parts of the output samples inherently includes the phase tracking function. (Note, this is a correct representation for an actual modem if the equalizer adapts more rapidly than the response time of the phase tracking loop in the modem.)
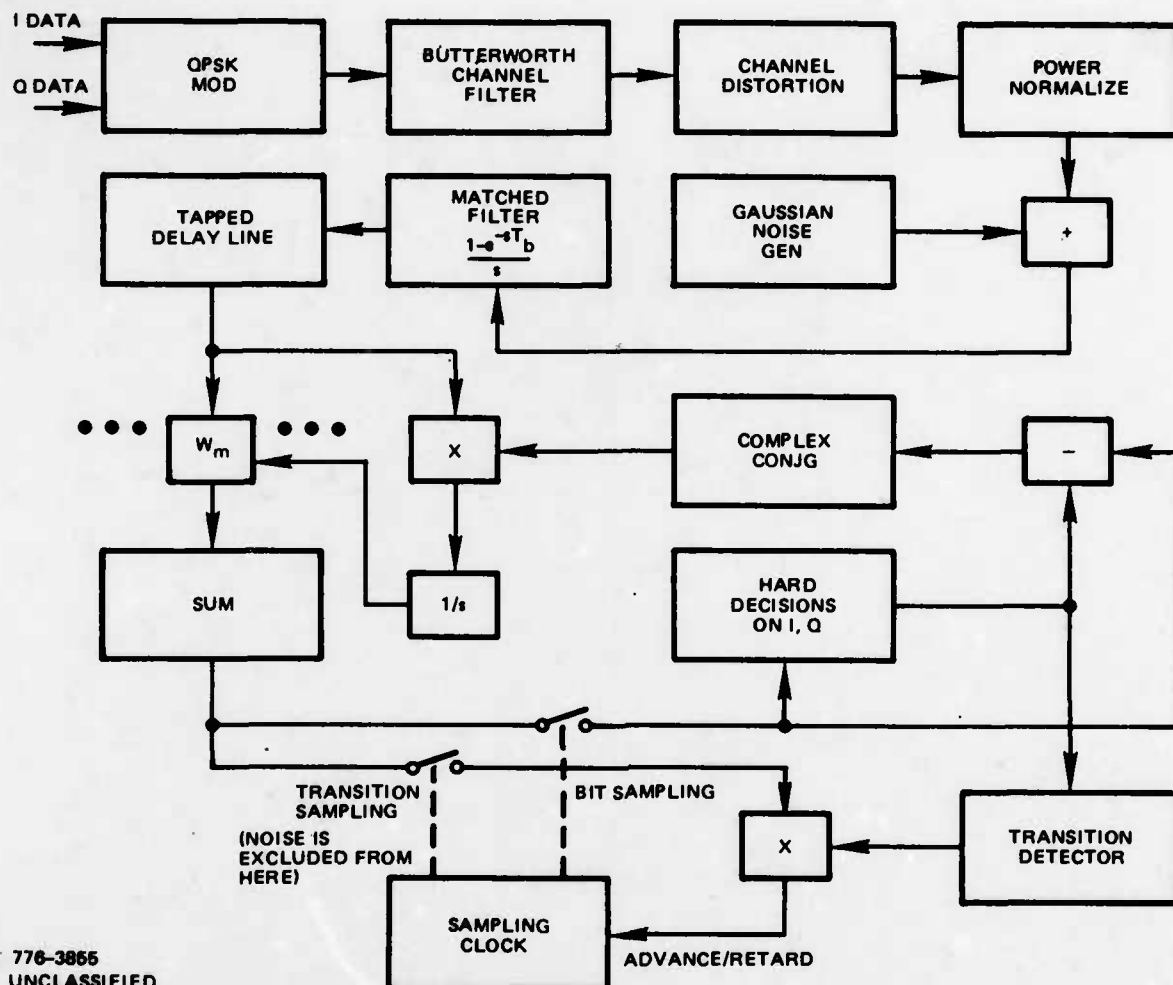
Figure 5-15. Computer Simulation of Equalizer Control From Modem Output
Samples – LMS Algorithm

Clock tracking is implemented in the simulation by sampling the equalized output at the bit transitions midway between the sampling times for the bit decisions. The transition sample is discarded unless there actually is a transition and the polarity of the sample is reversed for a negative-going transition. The real part of the transition samples are used to adjust the sampling clock phase by a feedback control. (Transitions in the imaginary part of the equalizer output are not used for programming simplicity.)

The simulation generates a periodic signal, typically 1024 samples with 32 samples per bit interval T_b, as the input to the equalizer. The LMS algorithm works with the bit decision samples in the period, holding a fixed clock phase for the full period. The transition samples are summed over the period, and the clock phase utilized for the next period is either advanced or retarded by one sample in accordance with the polarity of the sum.

5-25

A random QPSK data pattern is used repetitively; however, a new set of Gaussian noise samples is generated for each successive period. Noise is included in the equalizer adaption (both in the decision-directed error and in the tap outputs to the cross-correlation control), but is excluded from the transition samples applied to the clock tracking loop (since, typically, the delay-lock loop is made to have a narrow bandwidth to reduce jitter in the clock loop).

After a specified time for adaption, the equalizer tap weights are read out, and the average probability of bit error (hard decisions) is computed from the amplitudes of the bit decision samples. Since white Gaussian noise is presumed in the receiver, the noise power after equalization is computed by numerically integrating the magnitude-squared frequency response of the equalizer in cascade with the response of the matched filter. (An FFT is employed for this computation.) The bit decision amplitudes are obtained by passing the signal without noise through the equalizer with its final set of tap weight values.

The E_b/N_o is defined in terms of the signal power into the receiver. As seen in Figure 5-15, a power normalization is performed following the channel filter and distortion by averaging over one repetition of the signal. Thus, power lost in the transmitter due to a finite bandwidth filter is restored at this point in the simulation, and does not contribute to performance degradation.

5.2.2.2 LMS Algorithm Results

To test the capability to equalize from the bit decision samples out of the modem, channel distortion of the form

$$-.3\, e^{j1.25\omega T_b} + 1 +.3\, e^{-j1.25\omega T_b} = 1 - j.6 \sin (1.25\omega T_b) \tag{11}$$

was assumed.[*] Figure 5-16 plots the gain and phase corresponding to (11). Although phase distortion is of primary interest, and Figure 5-16 shows 62° peak-to-peak ripple, note that (11) has a small gain ripple also.

---

[*]The distortion is input to the simulation as the impulse response in the time domain.

776-3856
UNCLASSIFIED
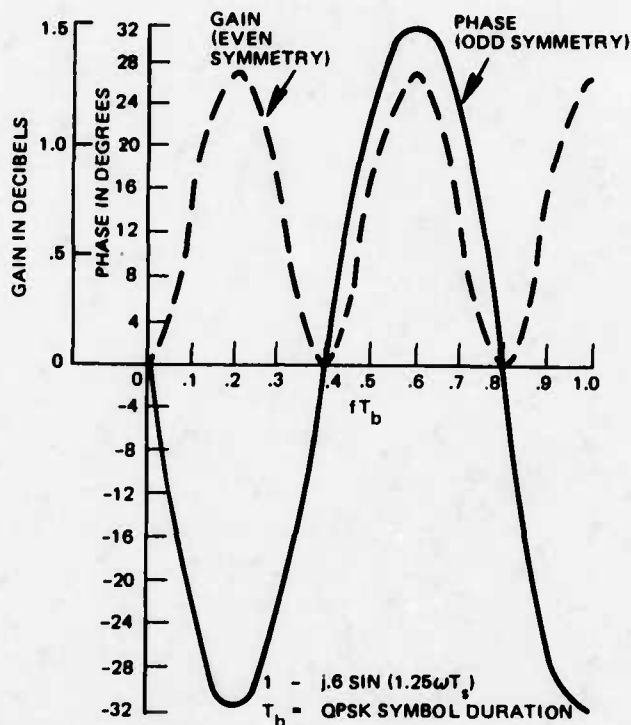
Figure 5-16. Channel Transfer Function

Figure 5-17 shows typical probability of error results, both with and without equalization. In the unequalized channel, the degradation increases with E_b/N_o because the bit amplitudes which are reduced by the intersymbol interference tend to dominate the average probability of error more at higher E_b/N_o. In contrast, with equalization the degradation tends to be constant over the indicated range of E_b/N_o, and is comparable to that obtained in Section 6 when controlling the equalizer from the entire waveform (rather than just with bit decision samples). Roughly the same performance is realized for tap spacing equal to either T_b or $T_b/2$.

Performance at $E_b/N_o = 1$ dB is studied because that approximates the operating point with rate -1/2 error correction coding.[*] Observe that decision-directed equalization still functions well at this point where the raw error rate is very high (7 percent). However, also note that the degradation is not very much to begin with at $E_b/N_o = 1$ dB. This illustrates the capability to tolerate more channel distortion when error correction is utilized.[**]

---

[*] With coding, the transmitted bits are redundant binary symbols, and we should write $E_s/N_o = 1$ dB. Then, with rate -1/2 error correction coding, $E_b/N_o = 4$ dB for the actual user-to-user data rate.

[**] Using the hard decision error rate to evaluate degradation gives an estimate of actual soft decision Viterbi decoder performance at the selected E_b/N_o (see Table 4-I). 5-27
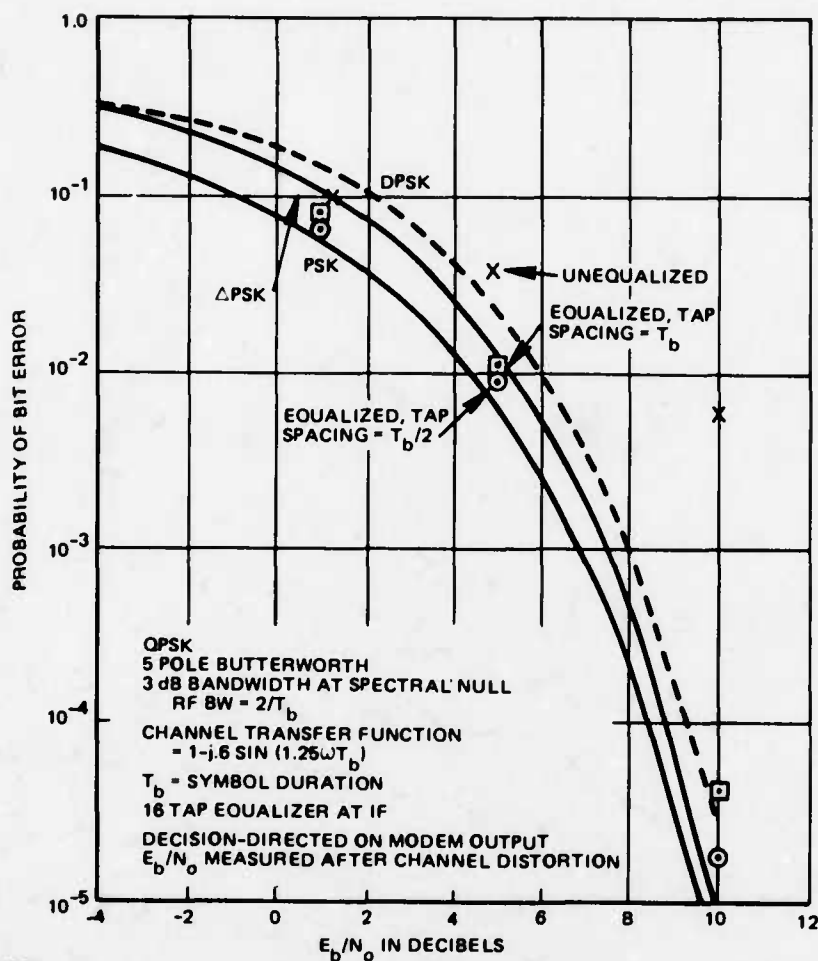
Figure 5-17. Results of Channel Equalization

Figure 5-18 gives similar results for more severe channel distortion of the form

$$-0.5 \, e^{j1.25\omega T_b} + 1 + 0.5 \, e^{-j1.25\omega T_b} = 1 - j \sin(1.25\omega T_b) \qquad (12)$$

Now the peak-to-peak phase ripple is 90°. The channel distortion is high enough to bring some of the bit decision amplitudes close to zero, and this explains why the error rate in the unequalized channel hardly decreases with E_b/N_o. After equalization, the degradation is relatively small, only slightly greater than obtained for the less severe channel distortion in Figure 5-17.
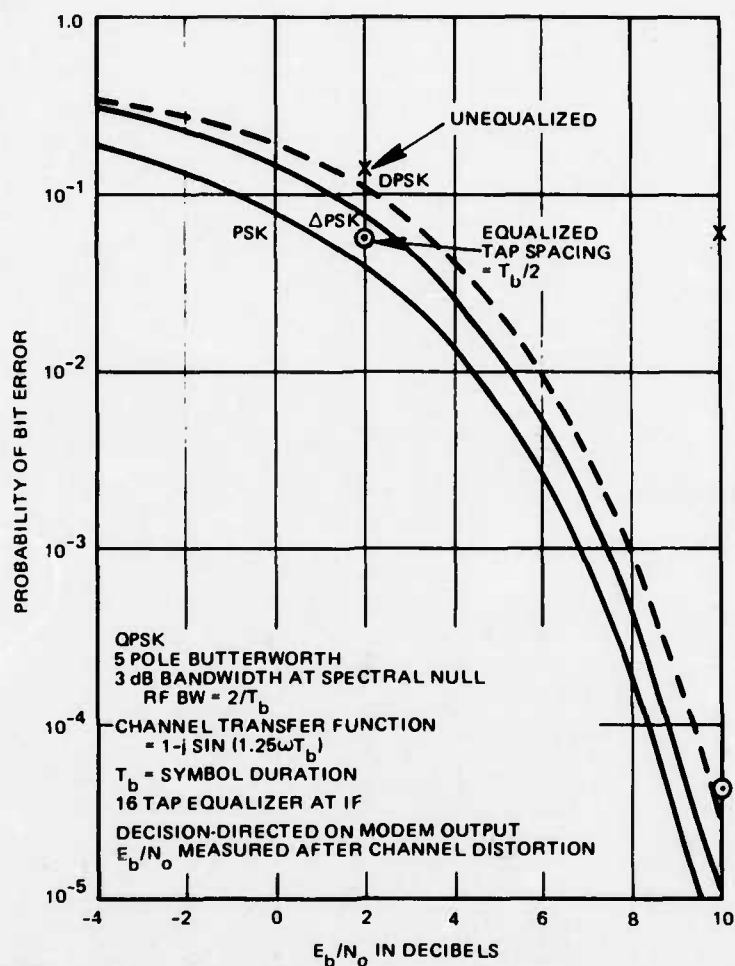
776-3858
UNCLASSIFIED
Figure 5-18. Results of Channel Equalization

5.2.2.3 Hard Limiting Channel

In connection with burst transmission for TDMA to the satellite repeater, the feasibility of equalizing in the receiver after hard limiting in the channel is of interest. It is presumed that the hard limiting follows the channel phase distortion; i.e., the phase distortion occurs in the transmitter or in the satellite repeater filters prior to hard limiting so that the equalizer cannot compensate for the distortion.

The computer simulation in Section 5.2.2.1 is easily converted to ideal bandpass hard limiting by power normalizing each complex sample, rather than power normalizing by averaging over a full period of the signal.

Assuming (12) for the phase distortion, and comparing with Figure 5-18, it is found that the error rate is about the same without equalization, but is higher after equalization. The equalizer is now relatively ineffective in correcting the phase distortion; however, there still is some benefit realized.

5.2.2.4   Dither Control of Equalizer Weights With Quantized Output From Modem

For purpose of investigating the feasibility of equalizer control from the 3-bit quantized output of the modem, a version of the dither technique to adjust the tap weights was implemented in the computer simulation of Section 5.2.2.1. While the actual implementation suggested by Figure 5-5 would probably employ a sequential dither of the real and imaginary parts of all tap weights, simulation of this would require excessive computer time. Instead, all tap weights are dithered simultaneously in the simulation.

The simulation procedure for dither control is now described. Evidently from Figure 5-15, the perturbation in any bit decision sample due to variation of a tap weight value is proportional to the corresponding tap output. Thus, for each bit decision sample, variation of all tap weight values may be performed, and the corresponding changes in the square error for that sample determined. The increment in the square error divided by the variation of the tap weight value is computed for each tap, and this gives finite difference approximations to the partial derivatives forming the gradient of the error. In the absence of modem output quantization, this procedure is equivalent to the LMS algorithm.*

The effect of modem output quantization is now apparent; namely, the variation of the bit decision amplitude due to dither of a tap weight is not observable in the output unless a quantizing threshold is crossed. To see how serious this problem can be, the structure of the quantizer must be defined. The real and imaginary

---

*The cross-correlation control shown in Figure 5-15 is the mathematical consequence of computing the partial derivatives of the square error with respect to the real and imaginary parts of the tap weights.

parts of the complex sample are separately quantized. Studies of Viterbi decoder performance [2] have shown that the quantizing levels should be spaced by approximately 0.5, where the standard deviation of the noise in the sample is normalized to unity, although a variation over an octave range causes little change in decoder performance. This result is in accord with the simulations shown in Figure 4-3. For this normalization, the amplitude of the signal is $\sqrt{2E_b/N_o}$, or 1.5 at $E_b/N_o = 0.5$ dB (E_b/N_o is defined for transmitted bits). This corresponds to $E_b/N_o = 3.5$ dB for data bits with a rate-1/2 code, implying operation at, or slightly below, an acceptable error rate (for soft-decision decoding). Figure 5-19 illustrates the modem output quantization as defined, assigning the midpoint between thresholds as the output level. At $E_b/N_o = 0.5$ dB, the signal amplitude is at the threshold for the output level with the highest magnitude.

For operation above $E_b/N_o = 0.5$ dB, the assumption is made for the simulation that the signal amplitude is held constant by the AGC and the standard deviation of noise decreases when E_b/N_o increases.

Although ideally the bit decision amplitude is set at a quantizing threshold according to Figure 5-19, the worst case with channel distortion is that all bit decision amplitudes lie between quantizing thresholds, so that the amplitude change due to dithering a tap weight is not observed in the absence of noise. However, operation with error correction implies the presence of noise with a standard deviation exceeding the quantization spacing by as much as a factor of 2. Thus, we expect the noise to tend to linearize the quantizer, in so far as controlling the equalization process, at least at low E_b/N_o.

In the simulation, a tap weight value (real or imaginary part) was dithered by ±0.1 (arbitrarily selected). Since the signal amplitude into the equalizer is set at unity (in-phase or quadrature data) in the simulation, a gain factor of 1.5 must be introduced by the AGC. The peak-to-peak change in amplitude due to the dither is thus 0.3, which is still smaller than the 0.5 spacing of the quantization levels. Consequently, equalizer control at high E_b/N_o will be poor. At low E_b/N_o, the linearization due to noise will improve the control.

Figure 5-20 shows how the 3-bit quantized model output is utilized to derive an output error. Internal to the modem is an AGC to hold the desired signal amplitude at the input to the quantizer. For purposes of the computer simulation, this AGC was implemented by feedback to hold 50 percent probability of getting the maximum
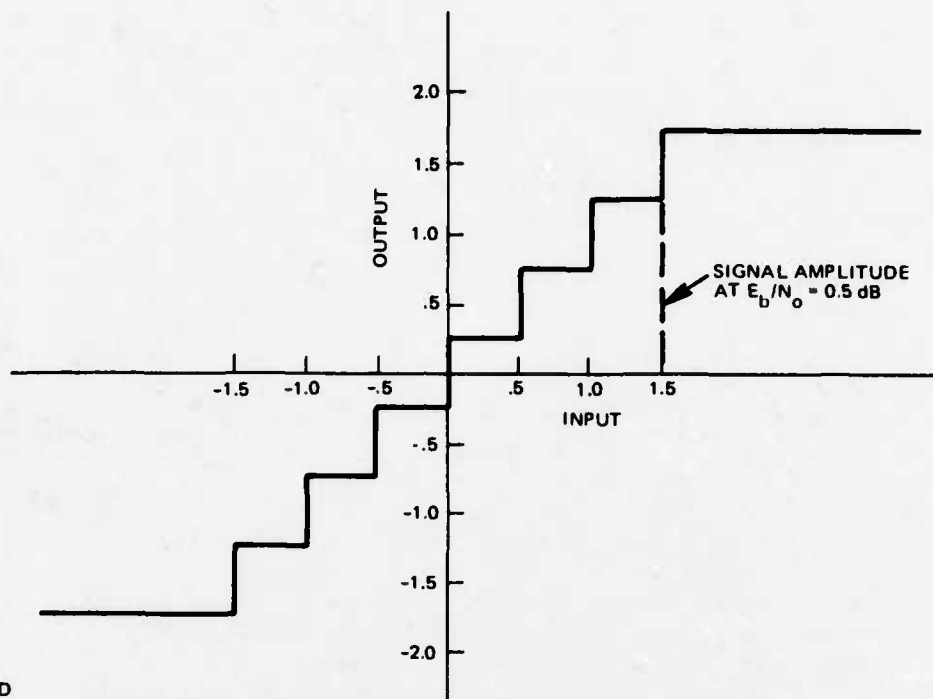
Figure 5-19. Optimum 3-Bit Quantization for Modem Soft Decisions
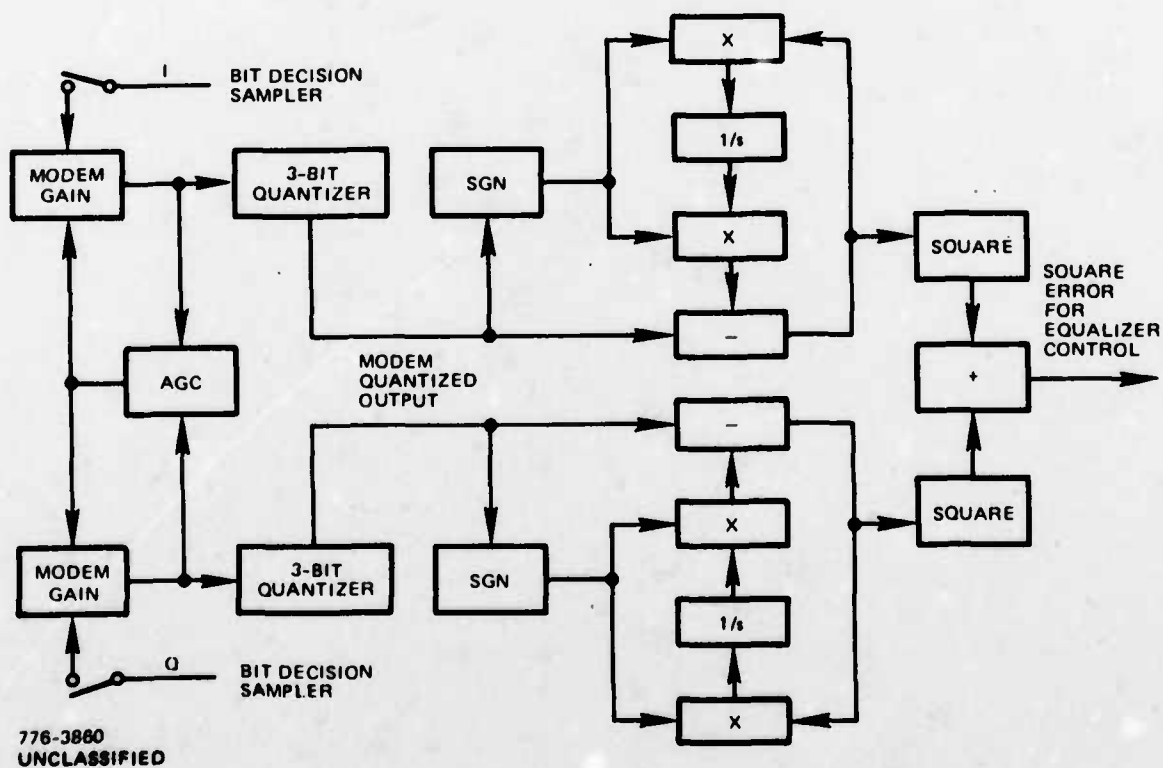
Figure 5-20. Decision-Directed Error With Modem Quantization   5-32

amplitude in the quantizer output. (This corresponds to holding an average signal amplitude equal to 1.5, as shown in Figure 5-19.) The modem gain is continually adjusted as equalization proceeds in the simulation.

Another control process is shown in Figure 5-20; namely, the assignment of magnitude to the hard decisions prior to forming the error. An LMS algorithm (i.e., cross-correlation) control is employed to optimize this magnitude; however, with the AGC described above, the optimum magnitude always turns out to be close to 1.5.

Dither control of the equalizer tap weights utilizes the square error as generated in Figure 5-20 from a bit decision sample. The real or imaginary part of a tap weight is first increased and then decreased by 0.1 and the corresponding changes in the bit decision sample are computed. The square errors for the two weight values are differenced to approximate the gradient, and the tap weight is proportionately changed. This is done for each tap weight, after which the process is repeated using the next bit decision sample.

5.2.2.5   Dither Control Algorithm Results

To test the dither control, the severe channel phase distortion of (12) was assumed. The probability of error after equalization by the dither control technique is plotted in Figure 5-21. Observe that good equalization is obtained for E_b/N_o in the range of 2 dB up to 5 dB. Comparing with Figure 5-18, it appears that the modem's 3-bit quantization has not impaired the equalization process at low E_b/N_o. However, relatively poor equalization is obtained at $E_b/N_o = 10$ dB with the modem gain setting as defined above to yield a signal amplitude equal to 1.5.

To be sure that modem quantization is the cause of impaired equalization at $E_b/N_o = 10$ dB, the computer simulation program with dither control was modified only to remove the quantization. As plotted in Figure 5-21, much better equalization is now obtained at $E_b/N_o = 10$ dB, almost equivalent to that previously found with the LMS algorithm (see Figure 5-18).

Next, the AGC was modified to hold 50 percent probability of getting the two upper levels in the quantizer output. From Figure 5-19, this corresponds to a reduction in signal amplitude to 1.0; thus, the spacing of the quantizing levels has been readjusted to 0.75 times the standard deviation of the noise at $E_b/N_o = 0.5$ dB. Theoretical decoder performance is only slightly worse for this increased spacing [2] and our results (Figure 4-3) corroborate this. After equalization with the 3-bit quantized output, performance is now as good as can be
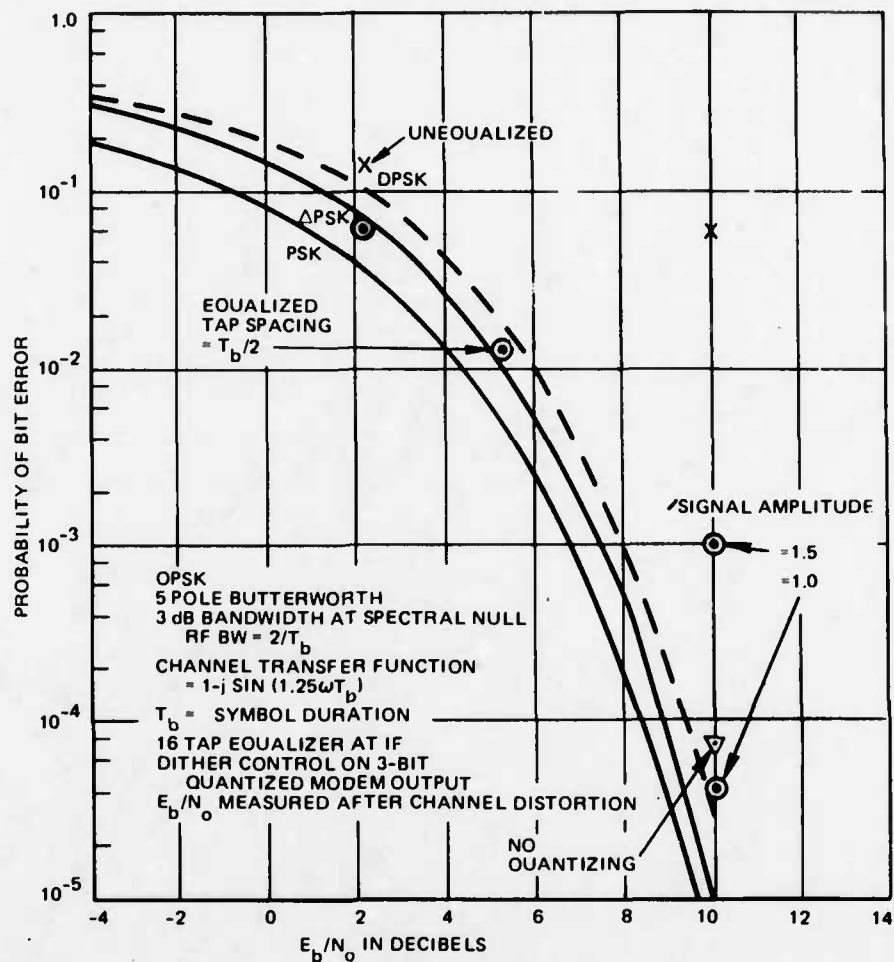
Figure 5-21. Results of Channel Equalization by Dither Technique

achieved with the unquantized output. (Figure 5-21 appears to show that controlling by the quantized output is slightly better; however, this presumably is due to a statistical fluctuation in the Monte Carlo simulation.)

5.2.2.6 Apparent Existence of Multiple Equalization Solutions from LMS Algorithm

An anomalous behavior of the LMS algorithm for equalizer control is described where the solution appears to depend on the initial values of the tap weights. Assuming correct bits (i.e., a training sequence), the square error after equalization is a quadratic form in the tap weights; hence, there must be a single set of tap weights minimizing the error. This unique solution ultimately will be found by any valid steepest descent technique. Presumably, the apparent anomaly in the LMS algorithm arises because the gradient of the error is close

5-34

to zero over a region of solutions, so that each iteration of its algorithm produces an extremely small change in the set of weights as soon as the solution falls within this region.

The computer simulation of Section 5.2.2.1 was modified to use the correct bits in computing the error. The correct bits are stored in an array and delayed so as to force the largest tap weights to be near the center of the tapped delay line equalizer. Since the correct bits are available to the LMS algorithm, an arbitrary set of tap weights can be used to start the adaptation procedure.* To obtain rapid convergence, the program was run with zero noise. Also, the assumed clock phase was held fixed by opening the delay lock tracking loop used to optimize clock phase.

When the tap weights seem to have reached a steady state condition, the values are read out, along with the integrated power response of the equalizer in cascade with the integrate-and-dump filter. In the absence of noise, the rms error after equalization is typically less than one percent; i.e., equalization is essentially perfect. The integrated power response therefore gives the E_b/N_o degradation for the resulting set of equalizer tap weights.** After equalization, the delay-lock tracking voltage is computed as described above. There are 32 samples in the QPSK symbol duration, and 32 symbols in the period.

When the tap spacing of the equalizer equals the symbol duration T_b, the LMS algorithm rapidly converges to the unique solution minimizing the error irrespective of the starting point. Of course, the solution is a function of the assumed clock phase.

The postulated channel distortion for the simulation is

$$1 - j \ .6 \ \sin(.25 \ \omega T_b) \tag{13}$$

---

*A decision-directed adaptation requires a starting point where correct decisions would be made in the absence of noise. Typically, one tap is nonzero to start the process.

**We expect that equalization by the LMS algorithm with noise present should yield slightly less degradation at any specified E_b/N_o, since the total error (intersymbol plus noise) is minimized for that level of noise.

where T_b is the symbol duration. Also, a 5-pole Butterworth filter with $BT_b = 1.0$ is present. (Filter bandpass equals twice the QPSK symbol rate.) Examples of the solution for 8 taps are given in Table 5-I. The tap weights are all real because the channel response is complex symmetric (gain is an even function of ω, phase is an odd function). Table 5-I shows that minimum degradation occurs for an assumed clock phase yielding a null in the delay lock tracking voltage. Furthermore, equalization does not defeat the action of the clock tracking loop to optimize the sampling instant, since there is a restoring force when the clock phase is displaced from the null point.

Table 5-I. Equalization with Tap Spacing Equal to QPSK Symbol Duration

5-pole Butterworth ($BT_b = 1.$). Channel Distortion = $1 - j\ .6\ \sin(.25\ \omega T_b)$			
	Assumed Clock Phase		
Tap Number	$.094\ T_b$	$-.094\ T_b$	$.281\ T_b$
1	.00	.00	.00
2	.00	.00	.00
3	.00	.00	.01
4	.03	-.02	-.17
5	1.16	1.05	1.11
6	-.25	-.04	.02
7	.05	.00	.00
8	-.01	.00	.00
Delay Lock Track Voltage	.53	.05	-.40
E_b/N_o Degradation	1.4 dB	0.4 dB	1.0 dB
RMS Equalization Error	.0068	.0041	.0059

Next, the simulation is carried out for 16 taps spaced by $T_b/2$. Again, the weights are expected to be real. Now, the solution appears to depend on the starting point. Table 5-II presents some examples. For an arbitrary assumed clock phase ($.094\ T_b$), the equalization process is started from all zero tap weights. Good equalization results, and the E_b/N_o degradation is small (although slightly greater than for tap spacing by T_b, despite the greater number of degrees of freedom). Then,

the assumed clock phase is shifted by $0.375\ T_b$ to $-.281\ T_b$, and the equalization process continues from the previous solution. After apparent convergence, a large delay lock track voltage is observed, and the E_b/N_o degradation is greater than previously obtained. The action of the delay lock tracking loop would drive the clock phase toward the previous assumption (from $-.281\ T_b$ toward $.094\ T_b$).

Table 5-II. Equalization with Tap Spacing Equal to Half QPSK Symbol Duration

5-pole Butterworth ($BT_b = 1.$). Channel Distortion $= 1-j\ .6\ \sin(.25\ \omega T_b)$			
	Assumed Clock Phase		
Tap Number	$.094\ T_b$	$-.281\ T_b$	$-.281\ T_b$
1	.00	.01	$.01+j.01$
2	-.01	-.01	$-.02-j.02$
3	.00	-.06	-.01
4	.04	.05	.01
5	.01	.16	.04
6	-.15	-.15	-.03
7	-.02	-.35	-.17
8	.56	.35	.11
9	.64	.80	.67
10	.05	.25	.51
11	-.17	-.09	-.06
12	.00	.02	-.13
13	.04	.01	.02
14	.00	-.02	.03
15	-.01	.01	-.02
16	.00	.00	.01
Starting Condition	All zero weights	Solution to left	All zero weights
Delay Lock Track Voltage	.01	-.40	.01
E_b/N_o Degradation	0.6 dB	1.4 dB	0.6 dB
RMS Equalization Error	.0061	.0088	.0077

Next, the equalization process is started from all zero tap weights for the same assumed clock phase, $-.281 \, T_b$. After apparent convergence, the solution is different than previously obtained and even has a slight imaginary component in two of the tap weights. Furthermore, the observed delay lock track voltage is essentially zero. The resulting E_b/N_o degradation is small, essentially identical to the first case where the assumed clock phase differs by $0.375 \, T_b$.

Mathematically, there can only be one correct solution for any assumed clock phase. However, there is not enough computer time available to observe the convergence to a unique solution, which must happen ultimately.

5.2.2.7   Conclusions

Use of bit decision samples from the QPSK modem to control equalization will enable decision-directed correction of severe phase distortion (as much as 90° peak-to-peak ripple could be equalized). Using the unquantized bit decision amplitude to form the error with respect to hard decisions, equalization is successful at both low E_b/N_o characterizing operation with error correction coding and high E_b/N_o for uncoded operation. However, the unquantized amplitudes are typically not an available modem output.

It appears feasible to utilize 3-bit quantized modem outputs to control the equalizer over the range of interest of E_b/N_o. Simulations verify that, particularly at low E_b/N_o, the noise tends to linearize the quantizer characteristic so that good equalization is obtained when controlling by quantized modem outputs. However, the AGC in the modem may have to be readjusted to ensure proper behavior at high E_b/N_o also. Viterbi decoder performance at low E_b/N_o is essentially unaffected by this readjustment.

The tap spacing of the equalizer is not a critical parameter. Essentially the same performance after equalization is achieved for the tap spacing equal to either T_b or $T_b/2$. With the latter, clock phase is less critical.

The anomaly where, depending on the starting point, different solutions apparently result from use of the LMS algorithm to minimize a quadratic form (the error) is presumably explained by an extremely shallow minimum when the error is viewed in multidimensional space with the tap weights as coordinates. The gradient of the error is essentially zero over a region of coordinate values, so that the LMS algorithm control produces extremely small changes within this region. The examples show that essentially equivalent performance is realized from the different equalization solutions.

5.2.3   UTILIZATION OF ERROR CORRECTION DECODER OUTPUT TO PROVIDE IMPROVED EQUALIZER CONTROL

The control of an adaptive equalizer implemented at IF can be derived from the error formed on the bit decision samples out of the QPSK modem which demodulates the equalized signal. For purposes of this equalizer control, it has been shown in Section 5.2.2.5 that the 3-bit quantized soft decisions available from the modem are essentially as good as the unquantized soft decisions, provided that the modem gain is set properly. This is an important practical conclusion when the unquantized output is not available on an existing interface.

Of course, the computation of the error necessitates knowledge of the true polarities of the transmitted data bits. Obviously, these true bit polarities are not available to the receiver unless a special training sequence is transmitted which can be exactly reproduced by a replica generator in the receiver. Since we desire to accomplish equalization on line using actual data, the bit polarities must be estimated in the receiver by a decision process.

The simplest method to estimate the bit polarities is by hard decisions directly on the modem output samples. The probability of error will be low at high E_b/N_o, and this decision-directed technique works well. The method is questionable at low E_b/N_o, corresponding to error correction encoded data, since there is a relatively high error rate on the modem hard decisions. However, computer simulations show decision-directed equalization to work very well over the E_b/N_o range of interest. (Minimum E_b/N_o = 2 dB for transmitted bits. With rate -1/2 error correction, this corresponds to minimum E_b/N_o = 5 dB for user-to-user bits.) Figure 5-20 functionally shows how the decision-directed error computation is made on the quantized output from the modem.

For uncoded data, hard decisions of the modem output give the only bit polarity estimate actually available, but then the error rate on this estimate is low ($\sim 10^{-5}$). With error correction where the modem output has a high error rate ($\sim.05$), the alternative exists of deriving a better estimate from the decoder output.

Then, the error computation shown in Figure 5-20 is modified to accept externally supplied bit polarities, as shown in Figure 5-22. A complication arises from the fact that there is an inherent decoding delay. Hence, the externally supplied bits are lagging and a compensating lag must be introduced on the modem output. Since the quantized modem output is digital, the lag is implemented by digital storage (shift register type with shifting rate set by the data clock from the modem), as seen in Figure 5-22. In terms of number of bits, the required storage is independent of data rate for a given decoder. A Viterbi decoder may typically have a delay of 32 bits. Also, some additional delay is introduced when the decoded data is passed through a replica convolutional encoder so as to reproduce the coded bits actually transmitted over the channel.

5.2.3.1   Comparison Between Control From Modem Hard Decisions and From Decoder Output

We can evaluate the potential benefit at low E_b/N_o of equalizer control from the decoder output by studying the extreme case where the externally supplied bits are forced to be correct. (Thus, we are really studying use of a training sequence.) The computer simulation described in Section 5.2.2.1 based on the LMS algorithm with error formed from hard decisions on the modem output is easily modified to force correct bits instead. One such modification consists of passing the signal through the equalizer without noise to arrive at correct decisions.* Then, the error on the noisy signal is computed with respect to these correct decisions, and used for the LMS algorithm control.

The assumed channel distortion is described by the transfer function $1 - j \sin(1.25 \, \omega T_b)$; in addition, there is a 5-pole Butterworth filter on the QPSK transmitted signal with 3-dB bandwidth equal to the transmitted bit rate. (Bandpass is twice the QPSK symbol rate.)

Results of equalization simulations at low E_b/N_o are given in Table 5-III. Although there is a very slight advantage obtained using correct bits instead of hard decisions, it is too small to warrant the additional implementation complexity. This conclusion presumes that $E_b/N_o = 0$ dB is the minimum value at which equalization control is needed.** This value is below the threshold of acceptable data with Viterbi decoding.

---

*The assumption is made here that intersymbol interference is not large enough to cause decision errors in the absence of noise.

**Equalization with hard decisions below $E_b/N_o = 0$ dB may still be feasible, but has not been studied since there is no need in the DSCS application.
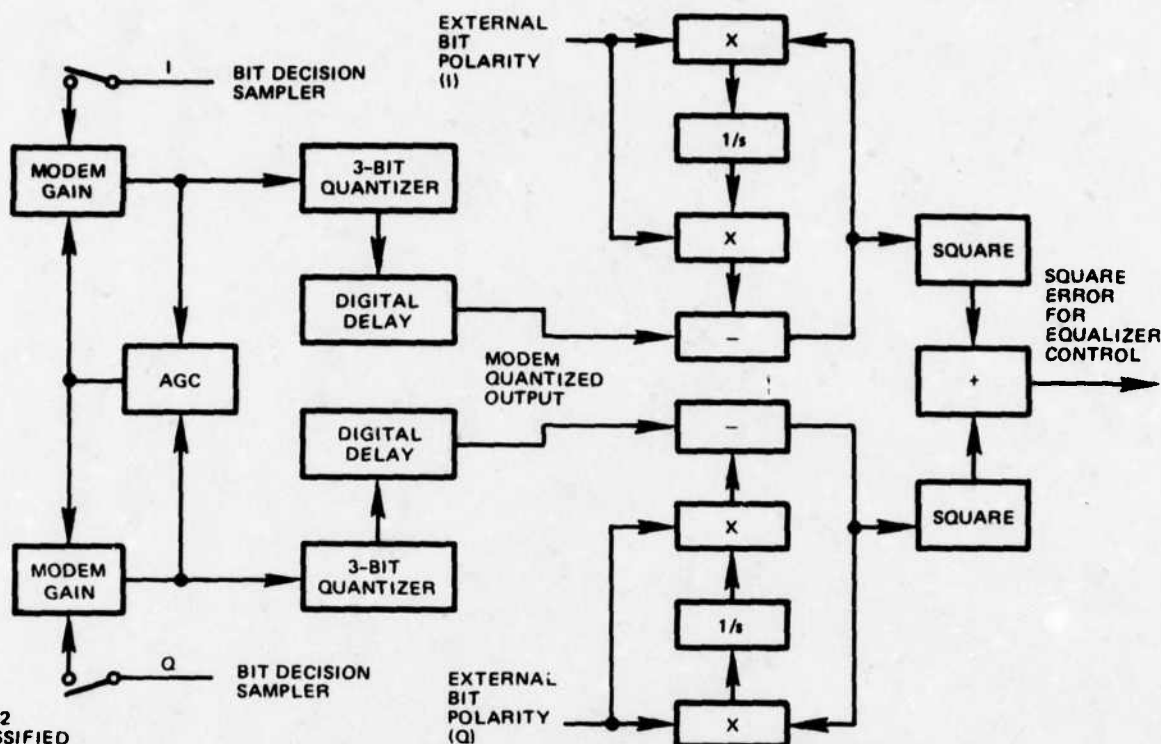
Figure 5-22. Externally-Directed Error With Modem Quantization

Table 5-III. Results of Equalization (Decision-Directed Versus Correct Bits)

5-pole Butterworth Xmit Filter ($BT_b = 1$). Distortion $= 1 - j\,\sin(1.25\,\omega T_b)$			
Error Computation	E_b/N_o	QPSK Error Rate	Degradation from Ideal QPSK
Unequalized	2 dB	0.141	4.4 dB
Hard Decisions	2 dB	0.056	0.9 dB
Correct Bits	2 dB	0.053	0.8 dB
Unequalized	0 dB	0.176	3.7 dB
Hard Decisions	0 dB	0.101	0.8 dB
Correct Bits	0 dB	0.095	0.6 dB

It should be noted that occurrences were observed where the decision-directed equalizer hung up on an incorrect solution, a phenomenon pointed out in Appendix D and explained as due to incorrect hard decisions even in the absence of noise. The incorrect convergence was eliminated by restarting the equalization process, temporarily at a higher E_b/N_o. Such excessive distortion is not likely to be encountered on the DSCS. Furthermore, it is found in Section 6.3 that using the decoder output does not provide significant additional protection against incorrect convergence.

776-3862
UNCLASSIFIED

5.2.3.2   Conclusions

        An adaptive equalizer can work on line as data is transmitted over the
channel. Decision-directed error computation from the 3-bit quantized soft decision
output of the modem provides usable control even at low E_b/N_o characterizing
operation with error correction coding. The additional complexity required to benefit
from the more reliable output of the decoder is not warranted by the negligible improve-
ment in equalizer control.

5.2.4   EQUALIZER CONTROL DERIVED FROM MODEM "ALPHA FLUNK"

        As discussed above, the control of the tap weights of an adjustable equalizer
can be derived from the decision-directed error on the bit decision samples (i.e.,
soft decision output) from the modem. Ideally, these bit decision amplitudes are
unquantized, and the error is a quadratic (i.e., square law) function of the difference
between the unquantized bit decision amplitude and the corresponding hard decision.

        If the soft decision output of the modem is quantized, typically to 3 bits,
a quantized error still can be formed. Feasibility of adaptive control from the
quantized error has been demonstrated already; however, at high E_b/N_o, a readjust-
ment of the quantizer spacing may be desirable to improve the convergence.

        The following describes equalizer adaptation for an even simpler error
quantity available from the QPSK/BPSK modem for DSCS.

5.2.4.1   Equalizer Control From Modem "Alpha Flunk"

        The QPSK/BPSK modem derives AGC from the bit decision
amplitudes, setting the gain so that the amplitude level of 1.0 (either positive or negative)
is exceeded half the time. Relative to this AGC-maintained level, the thresholds
for the quantized soft decisions are set at 0, $\pm$.25, $\pm$.50, $\pm$.75 to yield 3-bit
quantization. (See Figure 3-14.)

        "Alpha flunk" refers to failure of the bit decision amplitude (either
positive or negative) to exceed 0.5 amplitude. The average number of alpha
flunks is displayed on a meter, and a count of alpha flunks is available from
the modem. The original purpose for having the alpha flunk output is to monitor
modem performance.

        In simplest terminology, alpha flunk is equivalent to null zone
detection, where the null zone is below the 0.5 amplitude level.

5-42

For equalizer control, alpha flunk can be treated as an error quantity which is defined to be zero when the 0.5 amplitude level is exceeded and unity otherwise. The adjustment of the equalizer tap weights is, therefore, based on reducing the alpha flunk output of the modem to a minimum. Note that use of alpha flunk completely avoids need to compute a decision-directed error, in the sense that bit polarities are not of concern.*

5.2.4.2   Simulation of Equalizer Adaptation

The QPSK computer simulation described above in Section 5.2.2.4 for dither control of the equalizer tap weights was modified to derive control from alpha flunk. The AGC in the simulation adjusts gain by a feedback loop so that 50% of the bit decision samples exceed unity amplitude. (Note, the real part and imaginary part of each complex sample represent two bit decision samples for QPSK.) For each bit decision sample, the error is 0 if the sample amplitude exceeds 0.5, and 1.0 if the amplitude is less than 0.5. The real or imaginary part of a tap weight is dithered by ± 0.1 from its current value, and the resultant alpha flunks for the slightly larger and slightly smaller weight values are differenced to yield the control voltage to increment the tap weight value. This is done for all tap weights after each new bit decision amplitude is made available in the simulation.

To control gain, at each iteration all tap weight values are multiplied by a factor slightly greater or slightly less than unity, depending on whether the bit decision amplitude is less than 1.0 or greater than 1.0. Thus, the tap weights expand or contract, with a proportionate change in gain,** to maintain the level into the alpha flunk quantizer.

Figure 5-23 plots the variation of alpha flunk with time as the tap weights adapt for a case of severe channel phase distortion. The alpha flunk value is the average over one adaptation for a period of 32 samples, or 64 data bits for QPSK. Although there is a fluctuation due to the finite averaging time, the trend is for alpha flunk to decrease to the average expected for $E_b/N_o = 10$ dB. (See Figure 5-7.)

---

*Now, there is no possibility of improving convergence by utilizing known data bits or more reliable bits from the output of an error correction decoder, as discussed in Section 5.2.3.

**When the equalizer is used with an actual modem, one tap is held fixed to maintain a nominal output level into the modem.

DITHER CONTROL OF EQUALIZER TAPS
ALPHA FLUNK = FRACTION OF SAMPLES < .5
5 POLE BUTTERWORTH TRANSMIT FILTER
QPSK
16 TAP EQUALIZER
TAP SPACING = $T_b/2$
CHANNEL DISTORTION = $1 - j SIN(1.25 \omega T_b)$
E_b/N_o = 10 dB
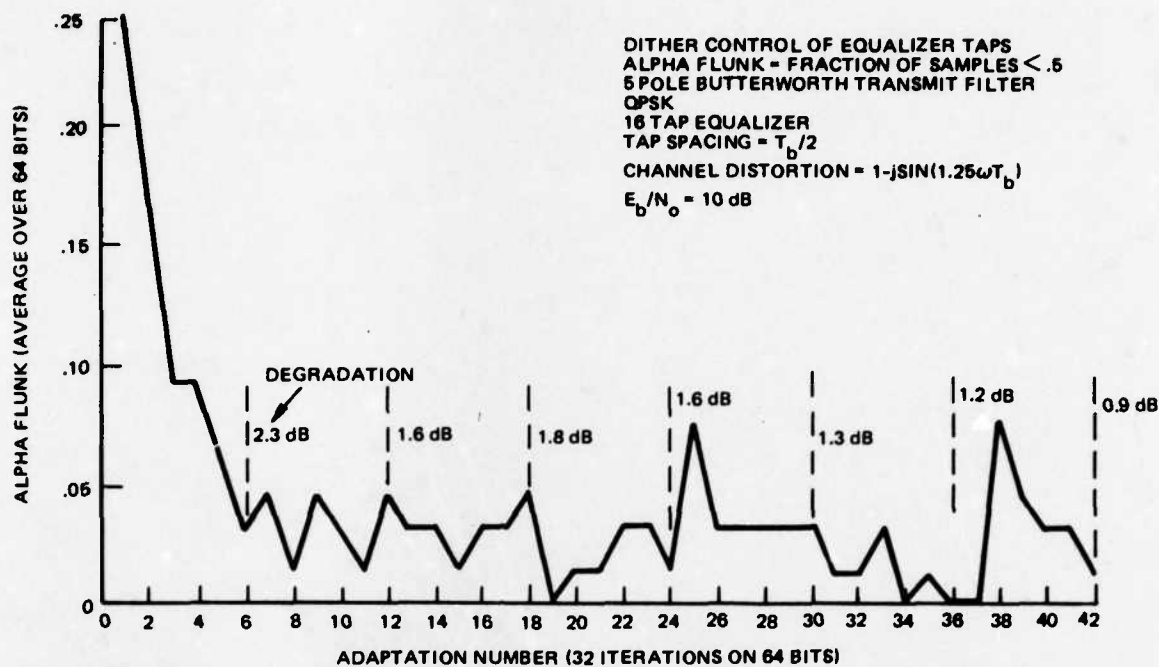
776-3863
UNCLASSIFIED

Figure 5-23. Typical Convergence Behavior With Control from Alpha Flunk

The E_b/N_o degradation is marked in Figure 5-23, and is computed after each six adaptations. The degradation is the reduction in E_b/N_o to achieve the same probability of error in an ideal QPSK system. Note that the adaptation with alpha flunk as the error causes the equalizer to converge to a solution having performance as good as previously found when equalizer control is derived from either the quantized or the unquantized decision-directed quadratic error (see Figure 5-21).

Similar results are obtained for adaptation at E_b/N_o = 2 dB. After convergence for the same channel distortion as in Figure 5-23, the degradation is 1.1 dB, essentially identical to previous results in Figure 5-21 for control derived from the decision-directed quadratic error. Possibility of converging to an incorrect solution was observed, as has previously been pointed out for decision-directed equalization.

5.2.4.3   Conclusions

The ability to derive equalizer control from the already available alpha flunk output of the modem is particularly convenient for manual adjustment. The equalizer, implemented as a tapped delay line at IF, has only the 70 MHz interface with the modem. The procedure for manual adjustment of the tap weights is to

reduce the alpha flunk output of the modem to a minimum. The taps are controlled sequentially, and it is probably desirable to adjust simultaneously the inphase and quadrature weight values for a selected tap, since it is impossible to eliminate coupling completely. (That is, changing the inphase weight for a selected tap will slightly change the quadrature weight, and vice versa.)

With the use of alpha flunk as the criterion for equalizer adjustment, there is no requirement to build an interface block to operate on the modem soft decisions, or for resetting the internal modem gain.

5.2.5   EQUALIZING MATCHED FILTER

This section presents simulation results for the system concept where the equalizer is incorporated within the modem. The equalizer simultaneously reduces intersymbol interference due to channel distortion and optimizes the detection process with respect to receiver noise. The concept is described above in Section 5.1.4. Since the equalizer control is by the LMS algorithm, the total mean square error due to both intersymbol interference and noise is minimized.

5.2.5.1   Computer Simulation Program

The system of Figure 5-9 was simulated with 32 samples per bit duration and QPSK data modulation. Periodic pseudorandom data sequences of 32 bits (independent sequences on the inphase and quadrature components) are employed; hence, the period is 1024 samples. Each sample is a complex representation of the bandpass signal, the real part corresponding to the inphase component and the imaginary part corresponding to the quadrature component. This signal is filtered by the presumed Butterworth transmit filter, with the signal centered in the passband. The filter has an adjustable bandwidth and number of poles. Then, the presumed channel distortion is applied, and the result is normalized to unit power. Independent Gaussian noise samples are added to the real and imaginary parts of all samples to represent additive Gaussian white noise, at a level corresponding to the presumed E_b/N_o in the receiver.

The receive bandpass filter is Butterworth with the signal centered in the passband. The 3-dB bandwidth and the number of poles are adjustable. Also, the number of taps and the tap spacing of the tapped delay line forming the equalizing matched filter are adjustable.

In accordance with the clock phase, the equalizer sum output with additive noise is sampled once every bit duration to obtain the complex bit decision samples. (There are 32 such samples in the period.) For each sample, the decision-directed error is computed by making hard decisions on the real and imaginary parts, and then subtracting these decisions to form the error e. This error is correlated with each tap output z, and each corresponding tap weight w is updated by the algorithm

$$w_{k+1} = w_k - K e_k z_k^*$$ (14)

where the subscript is a time index, K controls the time constant of the control loop, and the asterisk denotes taking the complex conjugate.

The initial clock phase is input to the simulation as a parameter. Sampling midway between the bit decision samples to get the bit transitions, the delay-lock error voltage is accumulated over one period without additive noise (taking just the real part for programming simplicity). For the next period, the clock phase is advanced or retarded by one sample in accordance with the polarity of the error voltage. In this simulation, noise is excluded from the clock tracking loop because the loop typically has a very narrow bandwidth which would be impractical for a computer simulation.

Initially, all tap weights are zero except at the center of the tapped delay line, where the taps over one bit duration are equal and nonzero, so as to start the decision-directed process. The equalization control process continues until convergence to a final set of tap weights has been obtained. Then, the bit decision samples over one period are computed for the final set of tap weights without additive noise, and the average probability of bit error is found for the presumed noise power.

It is necessary to compute the variance of noise on the real or imaginary part of the bit decision sample. Since white noise is added to the input, the overall frequency transfer function of the receive bandpass filter in cascade with the transversal filter determines the output noise variance. The square of the magnitude (i.e., power) of the overall response must be integrated with respect to frequency. *

---

*The power response need not be symmetrical about $\omega = 0$, which denotes the center frequency of the passband. Hence, the range of integration is $-\infty < \omega < \infty$.

The frequency transfer function of the equalizer with tap spacing T is given by (1).
The response of the equalizer can be computed easily at a discrete set of equally
spaced frequencies by applying a Fast Fourier Transform (FFT) algorithm. Taking
the square of the magnitude at each discrete frequency and multiplying by the power
response of the receive bandpass filter at that frequency, a numerical integration
over both positive and negative frequencies yields a good approximation to the noise
variance. In the program, a 1024-point FFT is utilized.

5.2.5.2   Simulation Results

All results are obtained for Butterworth filters on the transmitter
(i.e., in the channel) and in the receiver. The presumed channel distortion is
defined by (12). This distortion has a phase ripple of $\pm45^{o}$ and an amplitude ripple
of ±1.5 dB. Without equalization, the receiver is on the verge of making errors
due to the intersymbol interference.

The degradation is defined as the difference in decibels between the
actual E_b/N_o and the E_b/N_o which would produce the same average error rate in an
ideal QPSK system (e.g., error rate equals 10^{-5} at $E_b/N_o = 9.6$ dB in an ideal
system).

Table 5-IV presents simulation results for a variety of parameters, for
a channel both without distortion and with the distortion given by (12). Note that, in
the absence of channel distortion, the degradation with a transmit bandwidth equal to
twice the transmitted bit rate is relatively small, 0.9 dB, without the equalizing
matched filter when the receive bandwidth equals the bit rate. This is in good agree-
ment with published results for a sampling detector used with a receive bandwidth
equal to the bit rate[12]. The integrate-and-dump detector gives considerably more
degradation for this case of a narrow receive bandwidth. Naturally, the sampling
detector degrades when the receive bandwidth is widened; then, the benefit of the
equalizing matched filter shows up. If the receive bandwidth is narrower than the
bit rate, the degradation becomes significant.

In a practical implementation of the LMS algorithm, the correlation
between the error and the complex conjugate of a tap output required that one input
be hard limited. Simulations with hard limiting of the tap outputs displayed essentially
identical convergence behavior as with ideal multiplication.

Table 5-IV. Performance of QPSK Receiver With Decision-Directed Equalizing Matched Filter

Channel Distortion	Actual E_b/N_o	Transmit 3-dB Bandpass	Receive 3-dB Bandpass	Number of Equalizer Taps	Tap Spacing	Degradation
No Distortion	10 dB	$2/T_b$	$4/T_b$	8	$.25\ T_b$	0.2 dB
No Distortion	10 dB	$2/T_b$	$2/T_b$	8	$.25\ T_b$	0.2 dB
No Distortion	10 dB	$2/T_b$	$2/T_b$	8	$.5\ T_b$	0.3 dB
No Distortion	10 dB	$2/T_b$	$2/T_b$	8	T_b	1.4 dB
No Distortion	10 dB	$2/T_b$	$1/T_b$	No Equalizer		0.9 dB
No Distortion	10 dB	$2/T_b$	$1/T_b$	8	$.5\ T_b$	0.3 dB
No Distortion	10 dB	$2/T_b$	$1/T_b$	8	T_b	0.2 dB
No Distortion	10 dB	$1/T_b$	$1/T_b$	No Equalizer		2.6 dB
No Distortion	10 dB	$2/T_b$	$.75/T_b$	16	T_b	1.6 dB
No Distortion	10 dB	$1/T_b$	$1/T_b$	16	T_b	0.6 dB
No Distortion	10 dB	$1/T_b$	$.75/T_b$	16	T_b	2.5 dB
No Distortion	20 dB	$2/T_b$	$1/T_b$	8	T_b	0.5 dB
No Distortion	0 dB	$2/T_b$	$1/T_b$	8	T_b	0.4 dB
$1 - j\ \sin(1.25\ \omega T_b)$	10 dB	$2/T_b$	$2/T_b$	16	$.5\ T_b$	0.6 dB
$1 - j\ \sin(1.25\ \omega T_b)$	10 dB	$2/T_b$	$1/T_b$	16	T_b	0.7 dB
$1 - j\ \sin(1.25\ \omega T_b)$	10 dB	$1/T_b$	$1/T_b$	16	T_b	0.8 dB
$1 - j\ \sin(1.25\ \omega T_b)$	0 dB	$2/T_b$	$1/T_b$	16	T_b	0.7 dB

Transmit and Receive Filters are 5-pole Butterworth

5.2.5.3   Extension to Staggered QPSK

With SQPSK, a slight modification of (14) is necessary because the real
and imaginary parts of the error are sampled at a displacement of half the bit duration.
When sampling the real bit decision amplitude, the imaginary part of the error
in (14) is set to zero.* Similarly, when sampling the imaginary bit decision amplitude
at a displacement $T_b/2$, the real part of the error is set to zero.

Simulation of SQPSK with equalization by this modified algorithm showed
essentially identical results as given for QPSK in Table 5-IV.

An important attribute of SQPSK is that hard limiting after filtering does
not significantly raise the out-of-band spectrum splatter. The computer simulation
was additionally modified to hard limit the transmitted SQPSK signal after filtering,
with results given in Table 5-V, in the absence of channel distortion (other than the filters).
For each transmit bandwidth, the number of poles has been chosen to yield minimum
out-of-band spectral emissions[13].

Table 5-V. Staggered QPSK, Hard Limited After Transmit
Filter, No Channel Distortion

Transmit 3-dB Bandpass	Number of Poles	Receiver	Degradation
$2/T_b$	8	Unequalized	1.2 dB
$2/T_b$	8	Equalized	0.5 dB
$1.5/T_b$	6	Unequalized	1.5 dB
$1.5/T_b$	6	Equalized	0.6 dB
$1/T_b$	4	Unequalized	2.6 dB
$1/T_b$	4	Equalized	1.5 dB
Receive Filter is 5-pole Butterworth, 3-dB Bandpass = $1/T_b$ 8 Tap Equalizer, Tap Spacing = T_b			

Comparing Table 5-V with Table 5-IV, it is observed that hard limiting
of SQPSK after a narrow filter (bandpass = $1/T_b$) degrades performance. This is in
accordance with behavior noted previously, that equalization in the receiver is
relatively ineffective for channel distortion occuring prior to a hard limiter.

---

*This would be the approach for BPSK where the imaginary part is not sampled,
since it does not contain data.

5.2.5.4   Conclusions

Inspection of the results in Table 5-IV lead to a number of useful conclusions regarding the decision-directed equalizing matched filter for PSK signalling.

1.   Tap spacing should not exceed the reciprocal of the receive bandpass (Nyquist interval), to avoid significant degradation.

2.   Tap spacing less than the Nyquist interval negligibly improves performance.

3.   Receive bandpass can be as narrow as the transmitted bit rate for BPSK (one-half the total bit rate for QPSK) with negligible degradation.

4.   Transmit (i.e., channel) bandpass can be as narrow as the transmitted bit rate with small degradation.

5.   Extreme phase distortion in the channel can be corrected for all E_b/N_o in the range of interest ($E_b/N_o > 0$ dB).

The decision-directed equalizing matched filter represents a superior implementation concept for a PSK/TDMA modem which is to have P_e versus E_b/N_o performance close to an ideal system, particularly in an application where bandwidth constriction is unavoidable in the channel and/or the receive bandpass. The technique is applicable to BPSK, QPSK, or staggered QPSK. The equalizer control is by the LMS algorithm, based on the cross-correlations between tap outputs and the decision-directed error on the bit decision samples. The tap outputs can be hard limited into the cross-correlators, without affecting the convergence behavior.

5.3   DESIGN OF ADJUSTABLE IF TRANSVERSAL FILTER EQUALIZER

The preliminary design of a wideband 70 MHz IF transversal filter equalizer is presented in this section. The tap outputs of the tapped delay line are combined through variable bipolar attenuators which are controlled with adjustable bias voltages so as to yield the desired equalizer response.

The equalizer is intended to work with a BPSK/QPSK modem which has a maximum transmitted bit rate of 10 Mbps (with rate 1/2 coding, this is the transmitted bit rate on both inphase and quadrature components of the QPSK signal). Functionally, the equalizer is as shown in Figure 5-4. The inphase and quadrature components of each tap output are separately weighted to provide both amplitude and phase control.

For a maximum symbol rate of 10 Mbps, computer simulation studies in Section 5.2 have shown that the tap spacing can be as large as the transmitted bit duration. Consequently, a tap spacing of 0.1 microsecond is assumed for this preliminary design. A total of 8 taps is assumed since this number appears adequate to correct the likely channel distortion effects.

All components used in the design were chosen for wideband capability foremost above other requirements. At 70 MHz, only a small fraction of the operating bandwidth capability of most of the components is required in the 10 Mbps design. To operate with wider bandwidth modems, no change in components would be necessary, since the components have specification bandwidths of a minimum of 40 MHz in all cases. However, since the wider bandwidth modems will have correspondingly higher bit rates, a reduction in the tap spacing will be necessary. For 40 Mbps the delay line taps would have to be spaced a maximum of 25 nanoseconds, for example.

For this design study, a manually adjusted equalizer is presumed. With manual control, the tap weights are sequentially adjusted to reduce the alpha flunk error criterion available from the modem. It may be desirable to simultaneously control the inphase and quadrature weightings on a given tap, with sequential adjustment of the taps.

Because the tap weights are controlled by bias voltages, extension of the design to automatic control by a dither scheme should be straightforward.

5.3.1   BLOCK DIAGRAM

The block diagram in Figure 5-24 illustrates the major components in the equalizer and shows general signal flow throughout. The signal will be introduced at the left input to the delay line. Eight taps spaced 100 nanoseconds apart will be used to extract delayed samples of the signal before it is absorbed in the termination at the right end of the delay line. The tapped samples will be isolated and amplified in a set of eight wideband amplifiers and split into two inphase parts in subsequent 2-way power dividers. One of these parts will be weighted and combined with a corresponding part of each of the seven other taps. This combination is referred to as the I channel. The other part will also be weighted and combined with the corresponding parts of the other seven taps to form the Q channel. The I and Q channel sums of all taps will then be combined and given a relative phase shift of 90° in a quadrature combiner to form the equalizer output. By controlling the weights of
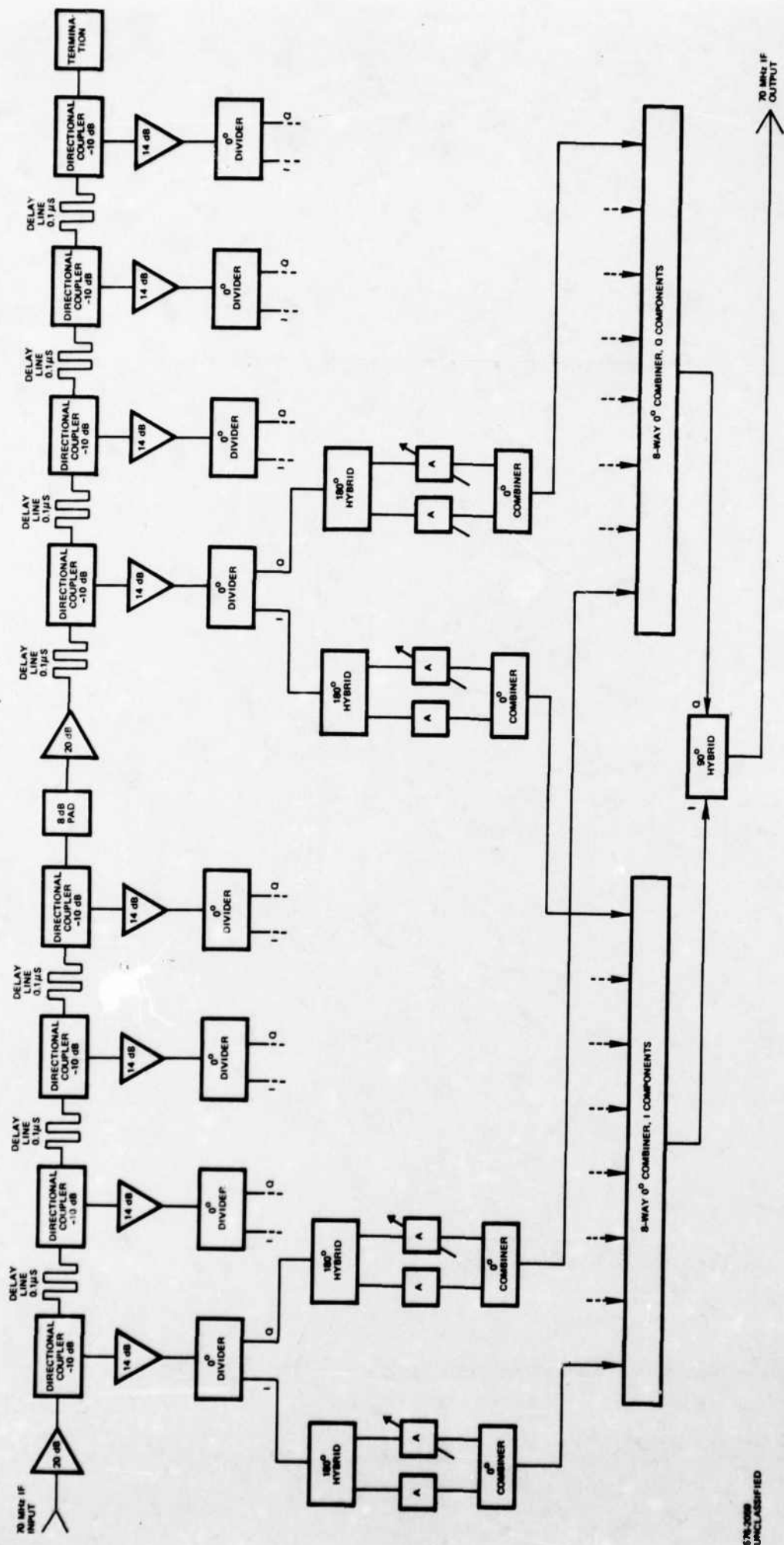
Figure 5-24. Block Diagram, Adjustable
IF Equalizer

individual I and Q tap outputs, each delay line tap can be effectively adjusted to any desired magnitude and phase before it reaches the output.

5.3.2 DETAILED DESIGN DISCUSSION

5.3.2.1 Tapped Delay Line

The tapped delay line will consist of seven lengths of coaxial line* and eight directional couplers. Figure 5-24 illustrates how the lines and couplers will be situated to provide the eight taps each separated from the adjacent tap by 100 nanoseconds. Each delay will be accomplished using 70 feet of coaxial cable. Semi-rigid coax, RG-402, will be used. Each 70 feet line will be coiled into a 3 inch diameter, 6 inch long spool with SMA connectors on the ends. The directional couplers will be used to effect isolated taps on the delay line. Ten dB coupling loss is a compromise between isolation and insertion loss. Low VSWR type couplers will be used to prevent reflections and subsequent phase and amplitude distortion.

As shown in the block diagram, the 70 MHz IF input signal will be amplified a nominal amount, sampled, delayed in a coiled cable, sampled again and so on until four taps have been formed. Four such taps are considered to be the optimum number, due to the buildup of VSWR and insertion loss, before some gain must be introduced to keep signal levels relatively high. Signal levels which are too low would be subject to noise and interference. A 10 dB fixed, low VSWR attenuator and a wideband, 20 dB gain packaged amplifier will be used to minimize VSWR and to bring the signal level back up to the starting point for the next four delay lines and couplers. The delay line will then be terminated in its characteristic impedance.

---

*An alternative design approach is to utilize surface acoustic wave delay lines, which have the advantage of reduced physical size. One design approach would fabricate the 100 nsec delays as individual SAW devices, replacing the individual lengths of coax. Alternatively the SAW device can be built as a tapped delay line. Study indicates that obtaining 800 nsec delay on a crystal of lithium niobate is well within the state-of-the-art. Propagation velocity is about 135 mils per microsecond, so the physical length is quite comparable to an integrated circuit type chip size. The only forseeable problem may be to provide enough physical and therefore electrical isolation between taps to prevent cross coupling. 100 nsec tap spacing is 14 mils on the device so a septum of some design would appear to be necessary to provide isolation. Experts from at least one company active in the production of SAW devices believe that such a device would not be a significant challenge. Estimated insertion loss would be about four to five dB due to the propagation (not including the loss of energy due to the tapping of one-tenth at each tap). Tap output impedance can be made low enough to work into a 50 ohm system similar to that described above for the coaxial delay line arrangement.

Figure 5-25 shows the nominal signal levels which will be encountered in the coaxial delay line approach. The typical RF terminal output level at 70 MHz is expected to be between -20 and -75 dBm in 50 ohms. A packaged wideband amplifier similar to the Avantek Corp. model UTO-512 will provide 20 dB nominal gain to raise the input to a maximum reference level of 0 dBm. The first coupler will have a true insertion loss, including coupled signal loss, of 0.8 dB. The RG-402 coax will have a 2.1 dB insertion loss for the 100 nsec length of line.

The signal level after the first coupler/line pair will be -2.9 dBm. After the fourth coupler the level will be -9.5 dBm. Approximately 10 dB net gain will be introduced to raise the level back up to the starting level of 0 dBm. The wideband amplifier will have an input VSWR of approximately 1.3, typically. The low VSWR fixed 10 dB pad will reduce that VSWR to about 1.1 at its input. Worst case VSWR at the first coupler input due to all devices in the string will be held to less than 1.5 to 1.

5.3.2.2   Phase and Magnitude Adjustments

Avantek Corp. model UTO-502 amplifiers will be used at each coupler output tap to provide further isolation to the delay line and to increase the signal levels in preparation for the weighting attenuators. Two-way power dividers with inphase outputs similar to Olektron Corp. model O-HJ-302V will be used to provide the dual outputs to be weighted and recombined. Divider loss is about 3.5 dB to each output. The entire chain being described is implemented with a 50 ohm characteristic impedance.

Since each tap output must be capable of being rotated 360 degrees in phase and varied in magnitude, the method shown in Figure 5-24 for weighting and phase shifting will be used. One inphase divider output will be designated as the inphase or I channel component. The other divider output will be designated as the quadrature or Q channel component. The channel designation refers to the port of the 90° hybrid combiner that each of these tap components will ultimately be fed to. Because both tap components, I and Q, are fed to the quadrature hybrid, the resulting output will be the vector sum. If the two are varied in magnitude with respect to each other, the resultant vector will vary in phase from zero to ninety degrees. If each I and Q component is not only varied in magnitude but also in sign, the resultant vector will vary in phase through 360 degrees and in magnitude from zero to maximum.
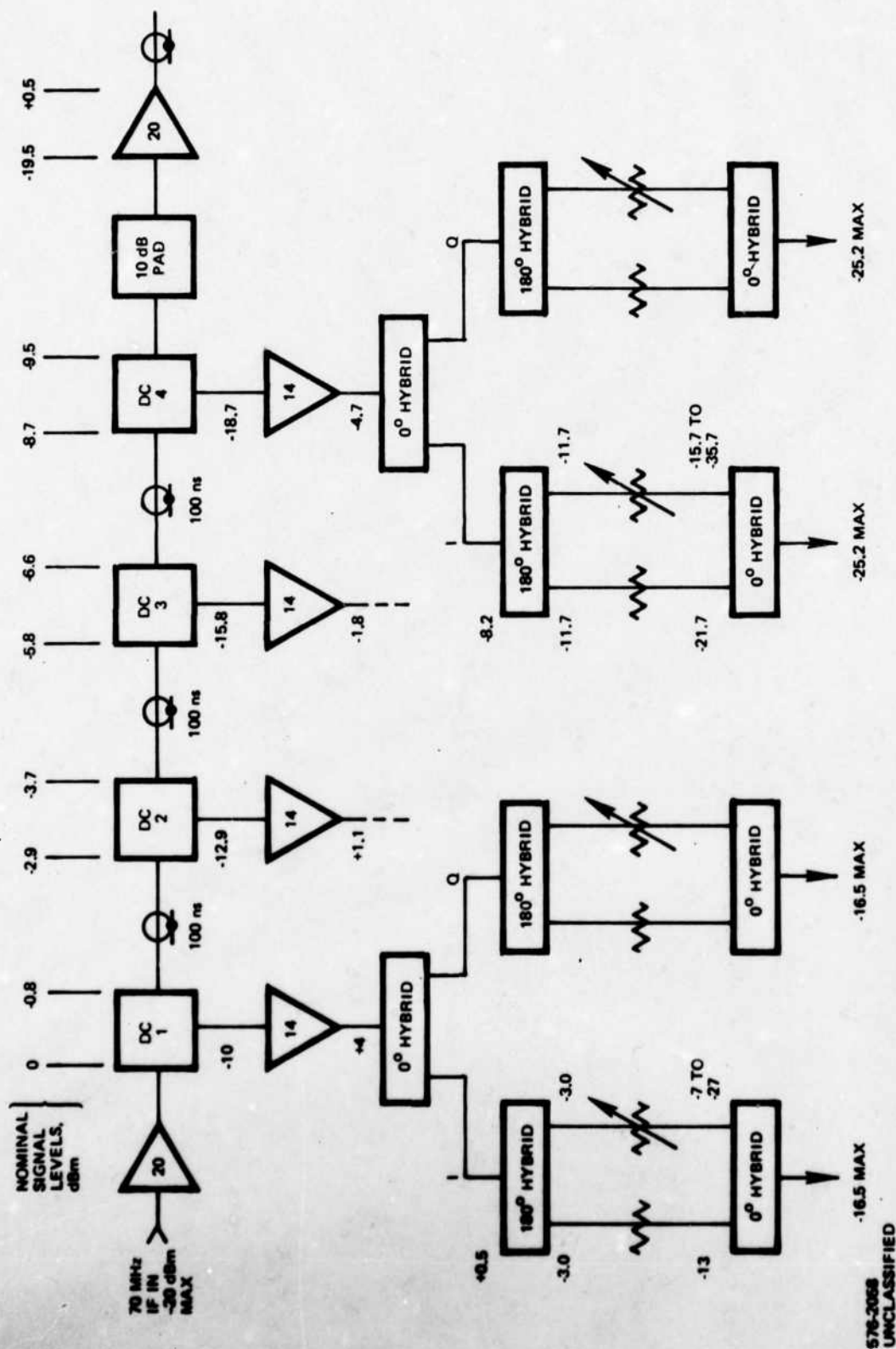
Figure 5-25. Signal Level Block Diagram

576-2058
UNCLASSIFIED

To accomplish this, each I and Q component will be split into two out-of-phase components in a 180° hybrid divider. One of the two outputs will be attenuated by a fixed amount using an in-line, 50 ohm attenuator. The other output will be attenuated a selected amount with a continuously variable PIN diode attenuator similar to the Watkins-Johnson G-1 unit. The minimum attenuation on the variable pad will be made six dB less than the fixed pad on the other 180° hybrid output. The two attenuator outputs will be summed in another inphase hybrid. As the variable attenuator is adjusted through its attenuation range, the resultant vector out of the inphase combiner will vary from maximum at zero degrees, pass through zero when both attenuators are equal, to maximum at 180 degrees. When the same is done to both I and Q components, the resultant at the quadrature hybrid will vary through 360 degrees inphase and zero to maximum in magnitude.

The G-1 attenuators are hybrid microcircuit devices housed in transistor cans. Each of the 16 attenuators in the equalizer will be housed in a connector type RF shield can for isolation between taps and tap components. The control bias for setting the loss of each attenuator will be set with one of the front panel potentiometers appropriately marked for each tap and tap component as shown in Figure 5-26.

5.3.2.3   Combining Technique

Referring to Figure 5-24, after the I component has been adjusted in magnitude and sign for each tap, it will be combined with the I components of the seven other delay line taps. An eight-way inphase combiner will be used to accomplish this. The Olektron Corp. model O-HJ-8-141 is a typical 8-way hybrid that can be used. The eight Q components will be similarly combined. Each of the functional elements in a given tap signal processing string has an associated trans-mission phase at 70 MHz. The worst case differential phase between such strings would be the sum of the differentials of each of the components. The transmission phase of these types of components is not normally specified and is controlled only by similarity of manufacture. To avoid large total differentials, the sixteen cables feeding the two eight-way combiners will be adjusted in length, during construction, to cancel any accumulations. If necessary, the same will be done for the two cables feeding the quadrature combiner.

As seen in Figure 5-25 the signal level at each inphase combiner output will be in the range from zero signal to a maximum of -16.5 dBm. Each 8-way combiner output with all inputs inphase and maximum would be as high as -11.5 dBm.

5-56
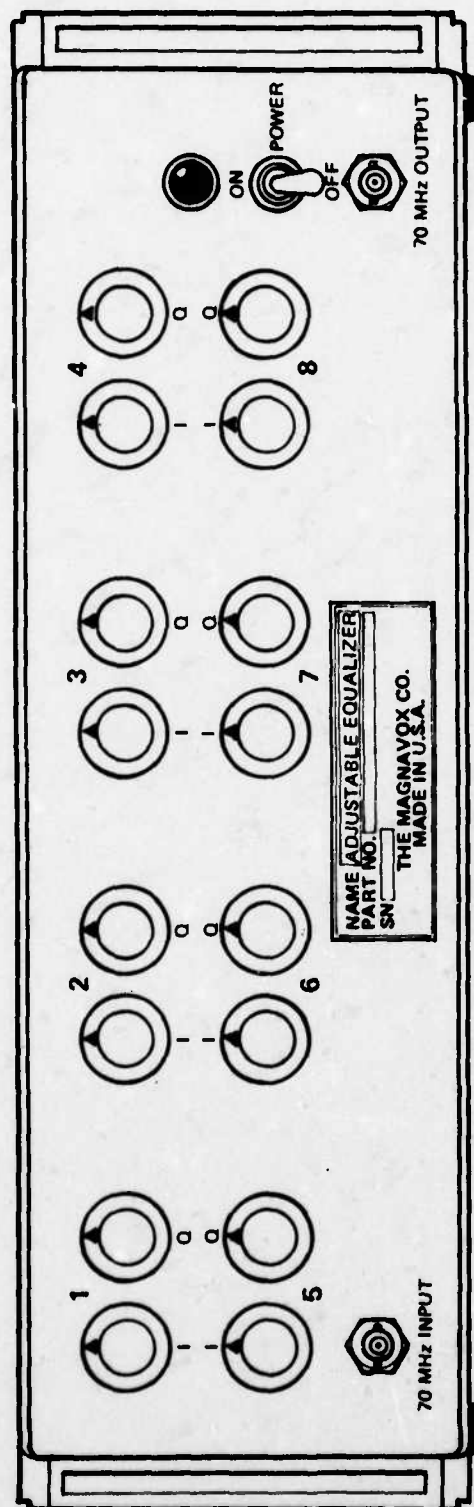
Figure 5-26. Front Panel View, Adjustable Equalizer

476-1714
UNCLASSIFIEO
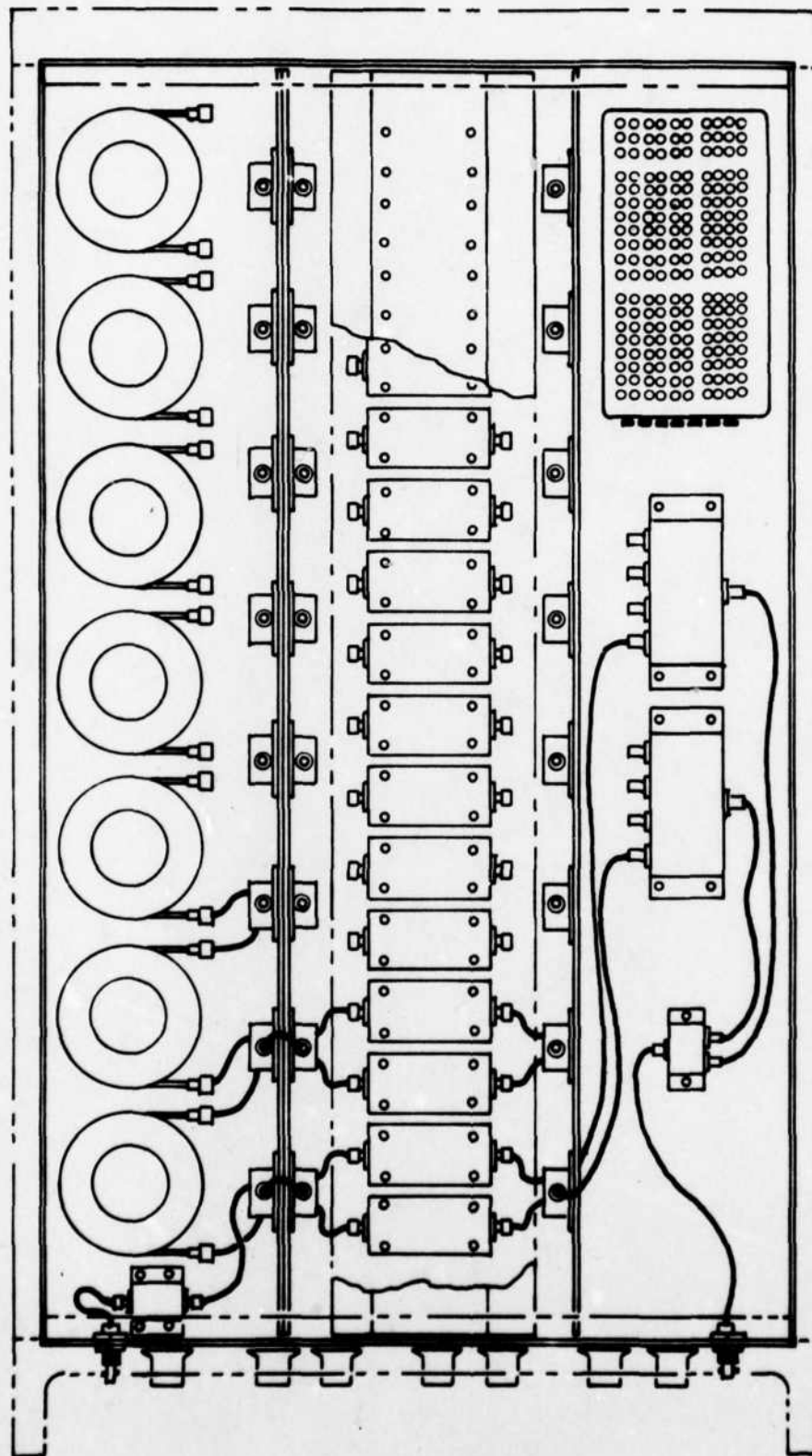
Figure 5-27. Top View, Adjustable Equalizer, Cover Removed

478-1713
UNCLASSIFIED

5-58

At these maximum levels, the quadrature combiner output would be approximately -12.5 dBm. However, any other combinations of weighting will yield lower output levels. The lowest expected level for one tap I or Q component at maximum would be -29 dBm. In any case the output level will be sufficient to drive the intended modem.

5.3.3   PHYSICAL DESCRIPTION

Figures 5-26 and 5-27 depict the approximate configuration of the equalizer front panel and chassis top view, cover removed. The entire adjustable equalizer including power supply is packaged in a rack sized chassis 7 inches high, 17 inches wide and approximately 25 inches deep. The only external connections are the 115 VAC 60 Hz power to the standard line cord and the 70 MHz coaxial input and output.

The chassis consists of standard Magnavox castings for the front and rear panels. The sides are a standard extrusion cut to length for this design. The top and bottom covers are simple sheet metal plates. Due to the multiplicity of small connector-type packages in the design, three vertical partitions are used as mounting surfaces. The PIN diode attenuator boxes are mounted to a long hat section member fastened to front and rear panels. The coiled delay lines are vertically mounted along the left wall of the chassis, and the directional couplers and isolation amplifiers mounted on the first partition facing the delay lines. Double shielded flexible coaxial cable assemblies using SMA connectors throughout are used to interconnect all 50 ohm ports. A single standard packaged power supply is used to supply the +15 VDC for the amplifiers and attenuators.

The front panel controls consist of a power switch with its associated power-on indicator and sixteen potentiometers for controlling the weighting for each I and Q component of each delay line tap.

SECTION VI
SYSTEM DEGRADATION DUE TO PHASE DISTORTION

Phase distortion in communications systems arises from several sources including filters, converters, and antennas. Depending on the magnitude and form of the phase distortion, the effects on system performance vary from minor to severe. This investigation quantifies the degradation produced for sinusoidal, cosinusoidal, and parabolic phase distortion in a SQPSK channel. It is shown that degradation is not solely dependent on the peak or rms phase error present but also on the type present and on the ripple frequency (for sinusoidal and cosinusoidal distortion). The effects of equalization and coding is determined for channels with phase distortion.

The system modelled is shown in Figure 6-1. It contains a five-pole ideal Butterworth filter with one-sided $BT_b = 1$. in the transmitter. The transmitter output can be hard limited if desired and the input PN sequences can be convolutionally encoded. The wideband receiver can include an adaptive equalizer and/or a Viterbi decoder following the integrate-and-dump detector.

6.1   DEGRADATION PRODUCED BY PHASE DISTORTION

For a linear bandpass channel without coding or equalization, the various forms of phase distortion produce the degradations shown in Table 6-I. Note that the phase distortion could be in either the transmitter or the receiver without changing the results in this linear system. Parameters SK and FR are discussed in Section 3.4. Results are presented for $E_b/N_o = 2$ dB and $E_b/N_o = 10$ dB. The results for $E_b/N_o = 2$ dB correspond to the range where coding would be used and give an estimate of the degradation expected with error correction.

These results show that a given level of peak sinusoidal or cosinusoidal phase distortion is far more detrimental to system performance than the same peak parabolic distortion. In fact, a very large parabolic distortion is required to produce a significant performance degradation. For example, at $E_b/N_o = 10$ dB, the transmit filter and 30° peak sinusoidal phase produce 5.09 dB degradation while the filter and 30° parabolic distortion produce only 1.11 dB degradation. Note that the average phase shift over the data bandwidth is only about one fourth the amount specified at the first null.
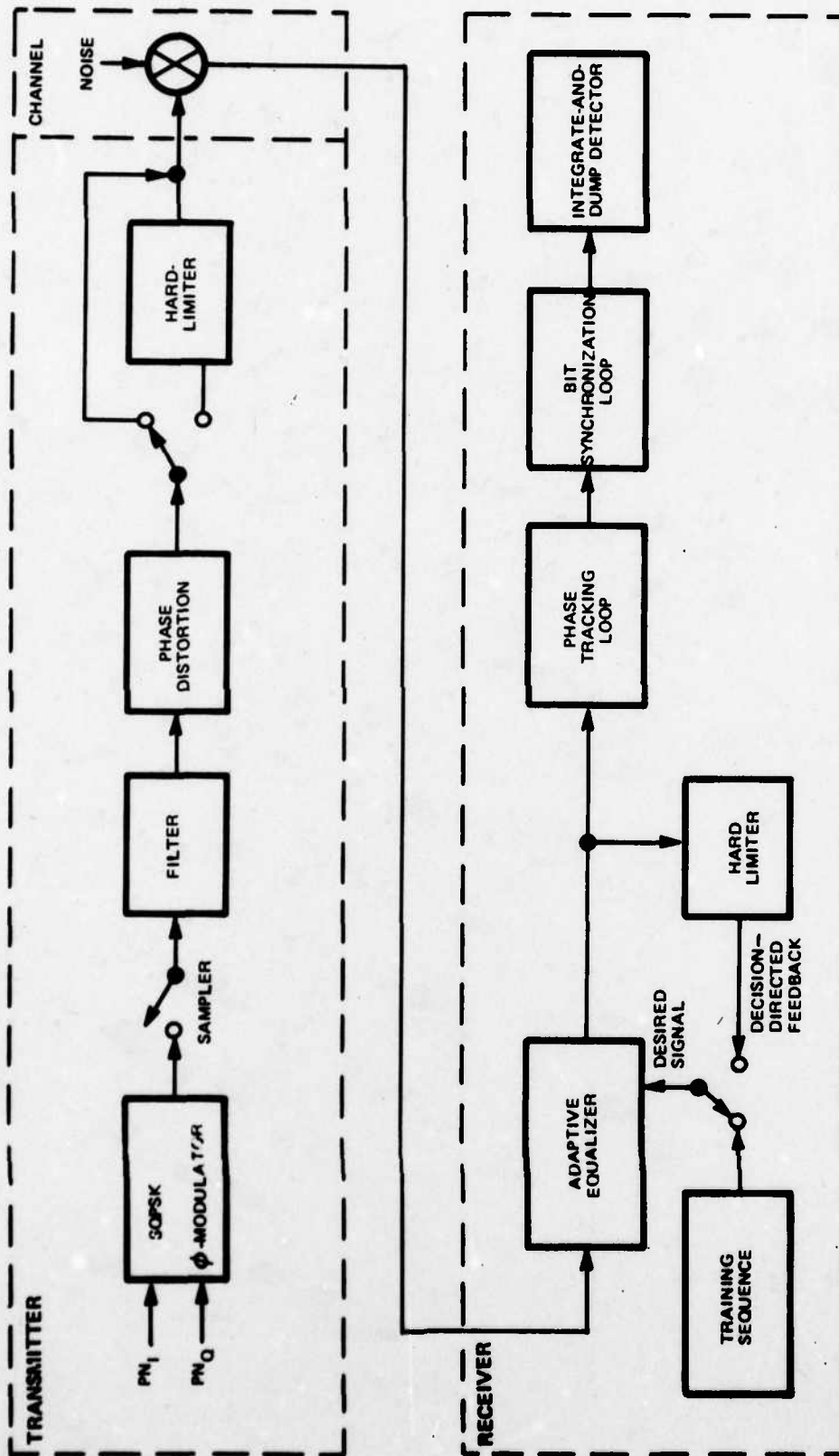
Figure 6-1. Communications System Modelled

Table 6-I. System Degradation Resulting from Phase Distortion and
Transmit Filter (5-pole Butterworth, $BT_b = 1$)

Type of Distortion	Parameters (SK, FR)	Degradation (dB)	
		$E_b/N_o = 2$ dB	$E_b/N_o = 10$ dB
No Distortion	(0, 0)	.47	.78
Sinusoidal	15, 1	.96	2.24
	30, 1	2.15	5.09
	45, 1	3.91	9.81
	15, 2	.53	1.16
	30, 2	.62	1.46
	45, 2	.92	2.26
Cosinusoidal	15, 1	.78	2.08
	30, 1	1.83	4.58
	45, 1	3.44	7.65
	15, 2	.58	1.32
	30, 2	.87	2.33
	45, 2	1.27	3.49
Parabolic	15, 1	.48	.99
	30, 1	.53	1.11
	45, 1	.61	1.28
	75, 1	.81	1.42
	90, 1	.95	1.60
	135, 1	1.20	2.18
	180, 1	1.61	3.05

As support for the results obtained, Figure 6-2 is included from reference 10 by J. J. Jones. (Note, BT = 2 in Jones means the same bandwidth as $BT_b = 1$, as defined here.) Several differences exist between the results in this report and Jones' results. While both simulations assumed QPSK data, Jones only considers the inter-symbol interference due to an adjacent bit while program SIMA computes the inter-symbol interference due to all bits in the sequence. In addition, Jones' curve does not show the overall degradation with both parabolic distortion and a transmit filter. However, he does show degradation due to cubic phase distortion and a system filter. As shown, the filter increases degradation at lower phase distortion magnitudes, but the degradation at higher phase distortion magnitudes is less with the filter than without the filter. This is mainly due to the fact that the filter attenuates the higher frequency components where the phase shift becomes very.large.
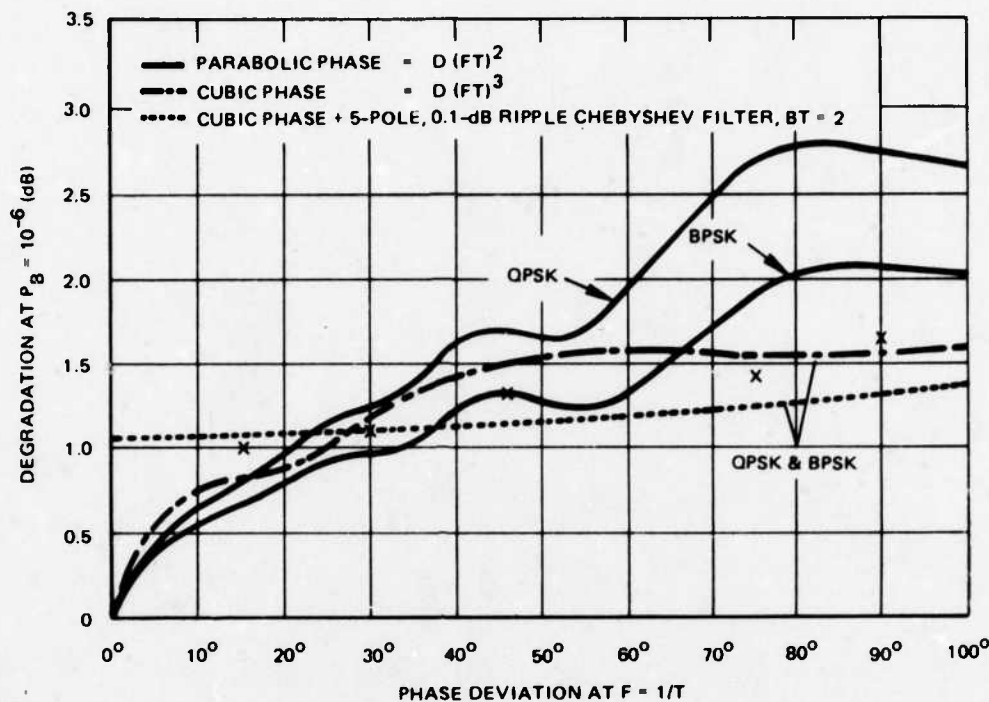
Figure 6-2. Degradation Resulting From Phase Distortion and/or Filtering (taken from reference 10). x Marks Degradation Obtained When QPSK Signals are Subjected to Filtering and Parabolic Phase Distortion.

Our results are plotted with Jones' curves to show, as he does, that relatively large parabolic phase distortion produces minor degradation compared to corresponding levels of sinusoidal or cosinusoidal phase distortion.

In setting system specifications on phase distortion, current practice involves specifying the maximum tolerable phase error. A better method would involve specifying the maximum phase error for each component of the phase distortion, whereby more parabolic phase error would be allowed than phase ripple (sinusoidal or cosinusoidal) in the system specifications.

In Table 6-I, degradation for cosinusoidal and sinusoidal phase distortion is shown for both FR = 1. and FR = 2. Note that the degradation is less for FR = 2. than for FR = 1. As FR increases the degradation decreases. Assuming a sinusoidal phase distortion

$$\theta(\omega) = \beta \sin(\omega T) \qquad (1)$$

or a cosinusoidal phase distortion

$$\theta(\omega) = \beta \cos(\omega T)$$

where T scales the phase ripple in the frequency domain, the response in the time domain after distortion contains paired echoes at delays of $\pm kT$ for $k = 1, 2, \ldots$ Since $FR = 1/T$ and the assumed bit rate is $T_b = 1$, the intersymbol interference falls directly on adjacent pulses for $FR = 1$. Similarly, for $FR < 1$, the paired echoes fall further away but the degradation remains the same as that degradation occurring for $FR = 1$. Therefore, the values shown in Table 6-I for sinusoidal and cosinusoidal distortion with $FR = 1$ are the worst case conditions for a given peak error SK.

6.2 EQUALIZATION OF CHANNELS WITH PHASE DISTORTION

Table 6-II presents the results for a linear uncoded channel having a transmitter filter followed by phase distortion with an IF equalizer in the receiver. $E_b/N_o = 10$ dB in Table 6-II. Two forms of the equalizer were used. The first assumes transmission of a known data training sequence (T.S.) which is used as the desired signal in the equalizer. The second type of equalizer hard limits the envelope of the equalizer output for use as the desired signal, as described in Section 5.1.2. This is a form of decision-directed feedback (DFB). The tap weight spacing is ISP = 16, which means that there are two tap weights per bit.

Table 6-II. Summary of Results (Linear Channel, ISP = 16, NTAP = 16, $E_b/N_o = 10$ dB)

Type of Distortion	Parameters (SK, FR)	Degradation Incurred (dB)	
		Equalizer with T.S.	Equalizer with DFB
Sinusoidal	15°, 1.	.85	.73
	30°, 1.	.86	.75
	45°, 1.	.87	.77
	15°, 2.	.72	.71
	30°, 2.	.73	.72
	45°, 2.	.74	.76
Cosinusoidal	15°, 1.	.86	.73
	30°, 1.	.86	.73
	45°, 1.	.87	.74
	15°, 2.	.86	.76
	30°, 2.	.88	.77
	45°, 2.	.92	.80
Parabolic	15°, 1.	.80	.98
	30°, 1.	.90	1.09
	45°, 1.	1.01	1.40

As seen by the results in Table 6-II, the adaptive equalizer improves performance of this linear channel in all cases. For sinusoidal and cosinusoidal distortions the improvement can be dramatic. For parabolic phase distortion, the equalizer with a T.S. yields a modest improvement while the equalizer with DFB produces essentially no improvement in performance. Note that the equalizer must cover a period of the input signal sufficiently long to contain all paired echoes of significant amplitude in order for the equalizer to effectively remove the effects of phase ripple. Since the paired echoes fall further away as FR decreases, an equalizer with a given tap spacing would require more taps to cover the increased delay of the echoes. Thus, the more rapid the phase ripple, the more total taps required in the equalizer structure.

Table 6-III considers the channel in which the filter and phase distortion are followed by an IF hard limiter. The resulting degradations are shown for the channel with no equalizer and for equalization with a training sequence (T.S.). No results are shown for the equalizer with DFB because the degradation would be the same as that for the channel without equalization. This is due to the fact that the hard limited input to the equalizer is the same as the hard limited output because the equalizer is initialized to have an impulse response with the center tap weighted at unity and all other tap weights are zero. Therefore the equalizer error quantity remains zero and no iteration of the tap weights occurs.

Table 6-III. Summary of Results (Hard Limited Channel, ISP = 16, NTAP = 16, E_b/N_o = 10 dB)

Type of Distortion	Parameters (SK, FR)	Degradation Incurred (dB)	
		No Equalizer	Equalizer with T.S.
Sinusoidal	15°, 1.	1.22	1.02
	30°, 1.	2.47	1.45
	45°, 1.	9.51	1.69
	15°, 2.	1.13	1.08
	30°, 2.	1.40	1.25
	45°, 2.	1.68	1.32
Cosinusoidal	15°, 1.	1.98	.73
	30°, 1.	3.68	2.28
	45°, 1.	6.28	3.67
	15°, 2.	1.24	1.13
	30°, 2.	1.88	1.37
	45°, 2.	2.81	1.89
Parabolic	15°, 1.	.99	.80
	30°, 1.	1.11	.92
	45°, 1.	1.25	.94

Compared to the linear channel, hard limiting acts to decrease the degradation of the unequalized channel but generally increases the degradation of the equalized channel. However, in all cases, the degradation is lower with equalization using a training sequence than the degradation with no equalization.

6.3 ERROR CORRECTION CODING FOR CHANNELS WITH PHASE DISTORTION

Convolutional coding of the input PN sequences was performed and a Viterbi decoder was fed the soft decisions from the integrate-and-dump detector. The serial coding mode was selected for SQPSK transmission and the coding parameters were rate = 1/2 and constraint length = 5. The results are presented in Table 6-IV. The first column gives the channel phase distortion parameters and five-pole Butterworth filter time-bandwidth product. The second column gives the coding parameters. The third column gives the received E_b/N_o at which decoding was performed in the Monte Carlo simulation. The fourth column gives the degradation that results from the Monte Carlo simulation. The fifth column tabulates the corresponding asymptotic coding degrading predicted by the minimum metric technique (Section 4.2) as $E_b/N_o \to \infty$. The last two columns give the degradations produced for the equivalent uncoded SQPSK channels for $E_b/N_o = 2$ dB and $E_b/N_o = 10$ dB respectively.

The first set of runs assumed sinusoidal phase distortion in the channel. In all cases, one cycle of ripple takes place over the frequency range from band center to the first data null (FR = 1.). Peak phase errors of 30° and 45° were assumed.

A run was also performed to evaluate the performance improvement with coded data resulting from use of IF adaptive equalization in the receiver.

The 32 received symbols, sampled 32 times per bit, are stored in an array X of size 1024. The 1024 samples of array X are shifted through the transversal filter portion of the equalizer with each shift generating an output time sample S. No noise is added to the input samples at this point. The equalizer adapts over a total of $12 \cdot 1024$ samples by treating the input array X as a circular array due to its periodicity. Finally, the tap weights are frozen at the final values obtained and the 1024 input samples are shifted through the transversal filter (with the LMS algorithm inactive) to form the array of equalizer output samples.

Table 6-IV. Degradations Produced for Coded and Uncoded Systems for **Channel Phase Distortions**

Phase Distortion Parameters	Coding Parameters	SNR	Viterbi Simulation	Asymptotic Degradation	Uncoded System E_b/N_o = 2 dB	E_b/N_o = 10 dB
Sinusoidal (SK = 30°) Filter (BT$_b$ = 1.)	Serial Mode K = 5	6	1.8	3.2	2.15	5.09
Sinusoidal (SK = 45°) Filter (BT$_b$ = 1.)	Serial Mode K = 5	8	3.7	5.12	3.91	9.81
Sinusoidal (SK = 45°) Filter (BT$_b$ = 1.) Equalizer (NTAP = 20, ISP = 16)	Serial Mode K = 5	4	.73	1.3	.68	.87
Sinusoidal (SK = 45°) Filter (BT$_b$ =) Equalizer (NTAP = 20, ISP = 16)	Serial Mode K = 5	4	.18	.19	.21	.27

NOTE: E_b/N_o and degradations are given in decibels.

Random noise samples are passed through the equalizer using the final set of tap weights obtained. The resulting output noise samples are correlated to some extent and the noise power is changed due to the transversal filter action. These output noise samples are appropriately scaled by the SNR value used and added to the detected soft decisions prior to the decoder input.

The Viterbi runs illustrate several important results. First, the Monte Carlo simulation results show that the degradation due to system distortions is close to that predicted by the degradation at $E_b/N_o = 2$ dB for an uncoded system. In addition, the asymptotic degradation, as predicted by the minimum metric technique, is an upper bound for all cases tested. Both techniques yield results in a far more economical manner than a Monte Carlo simulation and can be used to get an estimate of the degradation for coded data.

Coding does not appear to greatly improve the tolerance for phase distortion even for a fixed transmitted bit rate, where the user-to-user data rate is reduced by the coding. Fortunately, the use of adaptive equalization of the channel works effectively for soft decision decoding and greatly lowers the degradation produced by channel phase distortion. This is not surprising since the adaptive LMS algorithm used minimizes the rms error between the desired channel response and the equalizer output response.

These results indicate that coding improves channels with phase error as long as the hard decision bit error rate is not severe (as in the case of sinusoidal phase distortion with SK = 45°) and the reduced user-to-user data rate can be tolerated. Therefore, while coding does not conserve the spectrum (in fact, just the opposite), it is highly desirable where power must be conserved (since coding reduces the required E_b/N_o) and channel distortion is not severe.

SECTION VII
PERFORMANCE AND BANDWIDTH UTILIZATION OF FDMA

The scenario for the ground mobile forces assumes a large number of duplex FDMA signals. The total channel bandwidth is 45 MHz, and the data rate of each signal is approximately 50 kbps (100 kpbs total for duplex signals). One accessing carrier would typically time division multiplex 6 signals; however, this does not affect the computation of bandwidth utilization in terms of number of 50 kbps duplex signals in a specified bandwidth. It is highly desirable to pack as many signals as possible in the channel so that the bandwidth utilization factor is maximized subject to keeping adjacent channel interference to acceptably low levels.

This investigation has the goal of maximizing bandwidth utilization for FDMA with QPSK or SQPSK signaling. An equalized zero phase shift sharp cutoff filter is derived which permits a signal separation of as little as 1.1 times the data rate*in a linear channel, corresponding to an increase in capacity of about 25 percent as compared with the use of Butterworth filters having $BT_b = .75$, where T_b is the bit duration on either inphase or quadrature for QPSK. The filter performance is characterized and optimized for data rates varying about the nominal design point.

A large number of performance curves are presented for FDMA performance in saturating channels with either Butterworth filters ($BT_b = .75$) or equalized filters in the transmitter and receiver. For both the Phase II and Phase III (Hughes') TWT characteristics, signal separations of between 1.1 and 1.2 times the data rate are feasible with the equalized filters. The optimal backoff of the TWT amplifiers is found which minimizes the overall system loss in performance. A pure hard limited transmitter alters the transmitted spectrum sufficiently to negate the advantages of the equalized filter and signal separation is increased to about 1.5 times the data rate. Finally, the use of error correction coding produces much less degradation due to the effects of amplitude saturation and AM-PM conversion than the corresponding uncoded system permitting separations of 1.1 times the data rate with essentially the same degradation as at very large spacing. The minimum degradation occurs at signal separations of 1.3 times the data rate with equalized filters in the transmitter and receiver.

*As used here, data rate corresponds to uncoded BPSK. It is half the total data rate for uncoded QPSK.

7-1

7.1 EQUALIZED SHARP CUTOFF FILTER

This section presents the results of computer simulations to evaluate channel bandwidth requirements for operation of a QPSK modem incorporating integrate-and-dump detection (which is ideal only in a wideband channel). The previous study of a modem with an equalizing matched filter (Section 5.2.5) has shown the feasibility of 2 bps/Hz of bandwidth with E_b/N_o performance close to ideal QPSK. However, if the modem has an integrate-and-dump detector, degradation in an unequalized channel becomes significant when 1 bps/Hz is exceeded.

With an equalizer in the receiver, degradation of the modem due to a constricted bandwidth is reduced. The integrate-and-dump detector of the modem causes additional intersymbol interference when the channel bandwidth is narrow, but the equalizer acts to reduce this intersymbol interference by creating the inverse of the integrate-and-dump transfer function, at least over a narrow bandwidth (less than the null-to-null bandwidth).

An equalized zero phase shift sharp cutoff filter with $BT_b = .5$ is derived, its characteristics described, and the equalized channel performance presented. The variation in performance as a function of data rate in a fixed channel bandwidth is given to assess the sensitivity to deviations from nominal conditions. From these results, a compromise equalizer design is given where the design point sets the bandpass slightly wider than the bit rate.

7.1.1 BANDWIDTH REQUIREMENTS OF A QPSK/BPSK MODEM WITH INTEGRATE-AND-DUMP DETECTION

The computer simulation previously described in 5.2.2.1 with decision-directed equalization from the modem output is utilized to study the effect of a constricted channel bandwidth. A transmit filter is included in the simulation, and E_b/N_o is defined after the transmit filter. (Thus, power loss in the transmitter due to filtering is not counted as degradation.) The modem is represented as an integrate-and-dump detector, and a delay-lock loop controls the sampling time of the detector. Previously, the receiver was wideband; however, to study the effects of bandwidth constriction in the receiver, a receive filter is inserted following the point where white Gaussian noise is added. The adaptive equalizer in the receiver is adjusted by the LMS algorithm, deriving control from the cross-correlations between the decision-directed error on the modem bit decision samples and the filtered tap outputs of the equalizer.*

---

*In the simulation, the integrate-and-dump filter precedes the equalizer; hence, the tap outputs are properly filtered. In a practical implementation of this equalizer preceding the integrate-and-dump detector of the modem, dither control of the tap weights is applicable, as discussed in Section 5.2.2.

7.1.1.1   Results for Butterworth Filters

The simulation is run for several bandwidths, ranging from a bandpass of
$.75/T_b$ to $2/T_b$, where T_b is the bit duration (on in-phase or quadrature for QPSK).
First, the equalizer is deactivated, and only a transmit filter is present. Results
are plotted in Figure 7-1. These results for integrate-and-dump detection in an
unequalized, wideband receiver are similar to Figure 7 of reference 10. (Somewhat
less degradation is obtained here than shown in reference 10 because the error rate
is higher here and because E_b/N_o is defined in reference 10 prior to loss of power in
the transmit filter.)

Next, the equalizer is allowed to adapt from the decision-directed error
on the output samples from the integrate-and-dump detector. Results for the equalized,
wideband receiver are also shown in Figure 7-1. It is apparent that the equalizer has
improved performance considerably in the narrow bandwidth channel (less than $1.2/T_b$),
but has little effect* when the bandwidth is $2/T_b$).

Finally, a receive filter with the same bandwidth as the transmit filter
is inserted. Results are given in Figure 7-1 only for narrow bandwidths because the
receive filter has only a small effect for wide bandwidths. As may be anticipated,
the additional intersymbol interference due to the receive filter causes significant
additional degradation when the bandwidth is narrow and the equalizer is not active.
However, when the equalizer is allowed to adapt, it is found that the receive filter has
considerably improved performance at a bandpass equal to $1/T_b$. Remembering that
the transmitted signal is filtered to a bandpass of $1/T_b$, the explanation lies in the
noise reducing effect of the narrow receive filter, when the phase distortion is
removed by the equalizer. The result for integrate-and-dump detection with a band-
pass $= 1/T_b$ now is comparable to that found in Section 5.2.5.2 for the same bandwidth
and an equalizing matched filter, indicating that the equalizer does, in fact, compensate
for the integrate-and-dump transfer function.

When the transmit and receive bandpass are narrowed to $0.9/T_b$, the
performance even after equalization is considerably degraded.

---

*Of course, the equalizer would correct severe phase distortion, if it were present
 in the channel.

5-POLE BUTTERWORTH FILTERS
ACTUAL E_b/N_o = 10 dB
8-TAP EQUALIZER IN RECEIVER
TAP SPACING = T_b
QPSK WITH INTEGRATE AND DUMP
NO CHANNEL DISTORTION

UNEQUALIZED, WITH RECEIVE FILTER

EQUALIZED, WIDEBAND RECEIVER

UNEQUALIZED, WIDEBAND RECEIVER

EQUALIZED, WITH RECEIVE FILTER

DEGRADATION IN dB

3-dB TRANSMIT BANDPASS IN UNITS OF T_b^{-1}
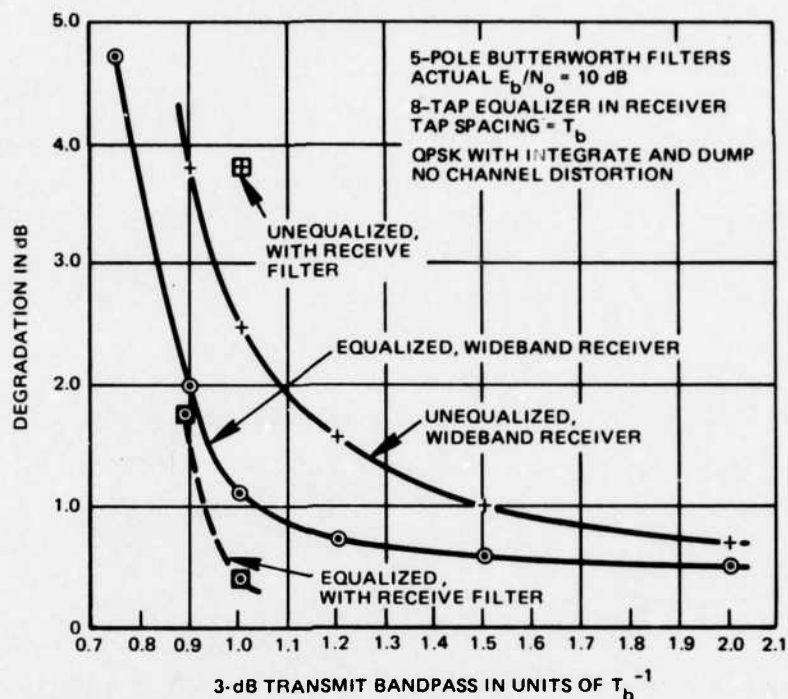
776-3971
UNCLASSIFIED

Figure 7-1. Degradation of QPSK With Bandwidth in Channel, Butterworth Filters

7.1.1.2   Results for Surface Wave Filters Without Phase Distortion

Next, the Butterworth transmit and receive filters are replaced by phase distortionless filters having a Butterworth amplitude transfer function. * Results are given in Figure 7-2. As may be expected, the cascade of a transmit filter and a receive filter is not much worse than having only a transmit filter, since there is no phase distortion. However, with a receive filter, the equalizer provides better performance, almost equal to ideal QPSK at a bandpass of $1.2/T_b$. Without equalization, performance is somewhat better than previously obtained for true Butterworth filters particularly when transmit and receive filters are cascaded.

A BPSK/QPSK modem with integrate-and-dump detection requires a relatively wide channel bandpass, in excess of $1.5/T_b$, to achieve performance reasonably close to ideal, assuming Butterworth filters. If phase distortionless filters are utilized (implemented as surface acoustic wave devices), somewhat less degradation results.

---

*In the computer simulation, this non-causal transfer function is easily applied in the frequency domain, after transforming by a FFT algorithm. In hardware, the realization is in the form of a nonrecursive (i.e., transversal) filter implemented as a surface acoustic wave device.
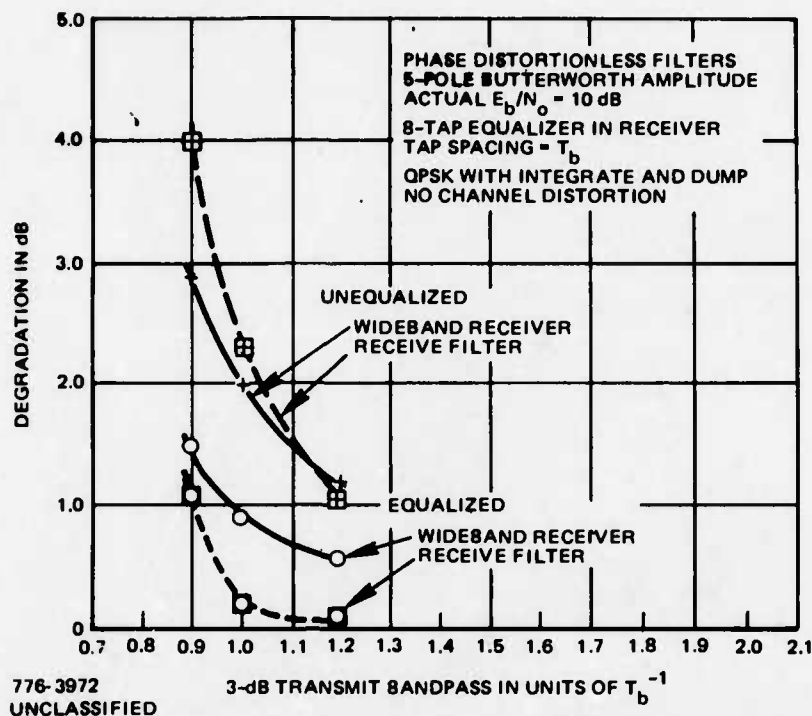
Figure 7-2. Degradation of QPSK With Bandwidth in Channel, Phase Distortionless
Filters

With adaptive equalization in the receiver, the degradation is reduced, and
the required bandpass is $1/T_b$. Phase distortionless filters with a Butterworth amp-
litude characteristic yield slightly less degradation than true Butterworth filters,
although the equalizer should remove the phase distortion of true Butterworth filters.
With implementation as a surface acoustic wave device, a specified equalized ampli-
tude characteristic is as easy to obtain as the maximally-flat characteristic.

7.1.2   QPSK AND SQPSK PERFORMANCE WITH EQUALIZED SHARP CUTOFF
         FILTERS

The degradation in E_b/N_o for narrowband filters is found to be substantial even
with phase distortionless filters. With a bandpass equal to T_b^{-1}, where T_b is the bit duration
(on inphase or quadrature for QPSK), the degradation is approximately 2 dB. With an
adaptive equalizer to combat the deleterious effect of a narrow bandwidth, the degra-
dation is reduced almost to zero for this same bandpass. It is found that the equalizer
acts to boost the channel response near the band edges, and this tends to force the
intersymbol interference toward zero.

7-5

7.1.2.1   Equalized Filter Response

If a phase distortionless filter with the amplitude response of a 5-pole Butterworth characteristic is presumed at both the transmitter and the receiver, the performance of a BPSK or QPSK modem with integrate-and-dump detection is degraded by 2.3 dB (at $E_b/N_o = 10$ dB*) when the 3 dB bandpass is T_b^{-1}. Incorporating an adaptive equalizer in the receiver reduces the degradation to only 0.2 dB, by boosting the response near the band edges to reduce intersymbol interference.

From the tap weights of the adaptive equalizer after convergence, the equalized receive filter response can be computed. This response (in decibels) is defined by

$$G(\omega) = H(\omega) - 10 \log_{10}[1 + (\omega T_b/\pi)^{10}] \tag{1}$$

where $H(\omega)$ is the transfer function (in decibels) of the adaptive equalizer itself. The second term on the right side of (1) is the Butterworth amplitude characteristic (in decibels). The transfer function $H(\omega)$ of the equalizer is defined by

$$H(\omega) = 20 \log_{10}\left[\frac{1.115 - .378 \cos(\omega T_b) + .128 \cos(2\omega T_b) - .046 \cos(3\omega T_b)}{.819}\right] \tag{2}$$

where the coefficients are given by the tap weights. All filters have zero phase shift.

Utilization of sharp cutoff filters at both transmitter and receiver is likely, in which case identical filters are desirable at both ends. For this case, the equalizer's transfer function can be split equally**(in decibels) between the transmit and receive filters. Figure 7-3 plots the filter response on this basis. Now,

$$G(\omega) = .5 H(\omega) - 10 \log_{10}[1 + (\omega T_b/\pi)^{10}] \tag{3}$$

where $H(\omega)$ is defined (in decibels) in (2). The passband is defined as the 3 dB point on the original Butterworth characteristic.

---

*Note, E_b/N_o is defined with respect to transmit power after the filter.

**The equal split does not theoretically optimize performance, but still gives excellent performance. The effort to iterate the design was deemed not worthwhile.

7.1.2.2   Simulation Results, Equalized Filter at Both Ends

The QPSK error probability was computed using the filter of Figure 7-3 at both transmitter and receiver. The degradation of 0.2 dB is essentially identical to that found previously using an adaptive equalizer and a Butterworth amplitude characteristic in the receiver, and a Butterworth amplitude characteristic at the transmitter. The bandwidth of T_b^{-1} implies 2 bps/Hz for QPSK and minor degradation from ideal performance.

Next, the sensitivity to deviation from nominal conditions was investigated by varying the data rate in a fixed filter. In the simulation, this was done by stretching the frequency scale of the filter transfer function in the frequency domain proportional to the assumed filter bandwidth (i.e., inversely proportional to the assumed data rate).

776-3973
UNCLASSIFIED
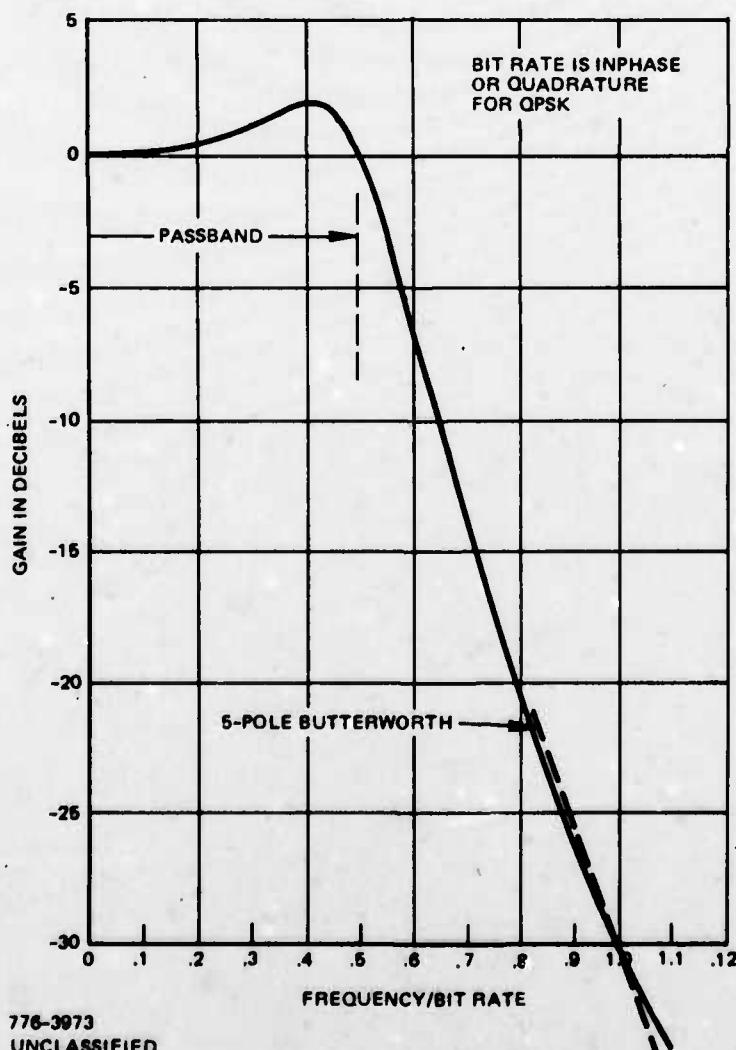
Figure 7-3. Equalized Filter for Transmit and Receive

Results are plotted in Figure 7-4. It is seen that increasing the data rate from nominal (i.e., reducing the bandpass below the bit rate) produces a rapid deterioration, consistent with the fact that the Nyquist rate is exceeded. However, decreasing the data rate produces a relatively small change in degradation. Initially, the degradation increases, to a peak of 0.8 dB at a bandpass of $1.2/T_b$, as the data rate is decreased. With a further increase of bandwidth, the degradation becomes negligibly small. Note that the lowpass equivalent filter with one-sided bandwidth-time product $BT_b = .5$ corresponds to a bandpass/bit rate of 1.0.

7.1.2.3 Simulation Results, Equalized Filter at One End Only

An alternate mode of operation would place a filter at either the transmitter or the receiver, but not both. The adaptive equalizer was allowed to converge with a Butterworth amplitude characteristic in the receiver, and no filter in the transmitter. Figure 7-5 plots the response of the Butterworth characteristic in cascade with the equalizer. The variation of performance as the data rate is varied is given in Figure 7-4. Again, performance deteriorates rapidly as the bandwidth is decreased so that the Nyquist rate is exceeded. However, as the bandwidth is increased, the performance changes little.

PHASE DISTORTIONLESS FILTERS
EQUALIZED AMPLITUDE RESPONSE
QPSK, INTEGRATE AND DUMP
$E_b/N_o - 10$ dB
BIT RATE IS INPHASE OR QUADRATURE

RECEIVE AND TRANSMIT FILTERS

RECEIVE OR TRANSMIT FILTER ONLY
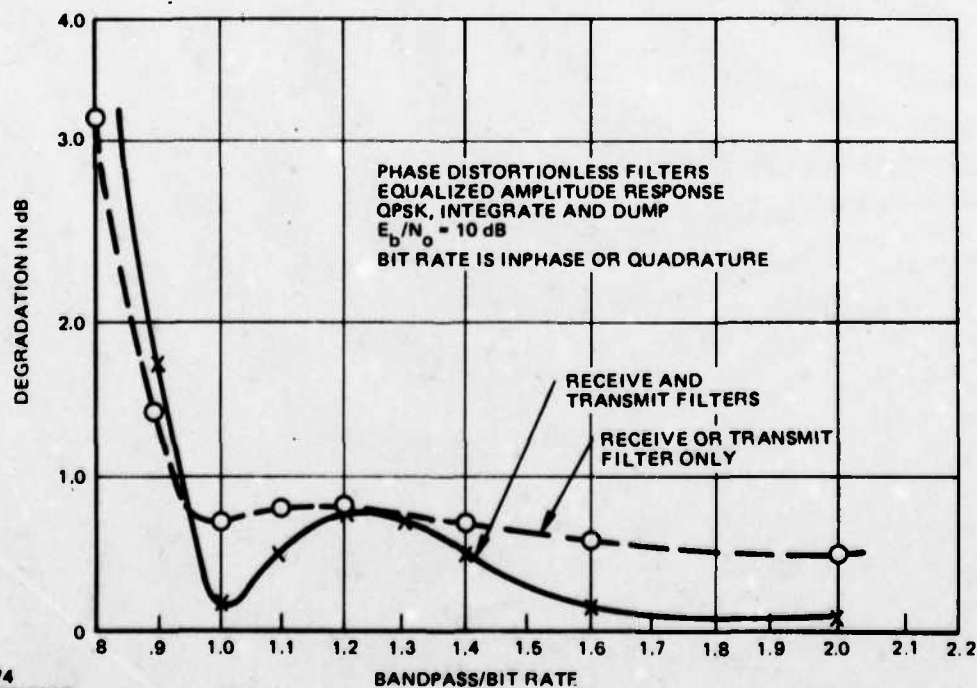
776-3974

Figure 7-4. Degradation With Varying Data Rate in Fixed Filter
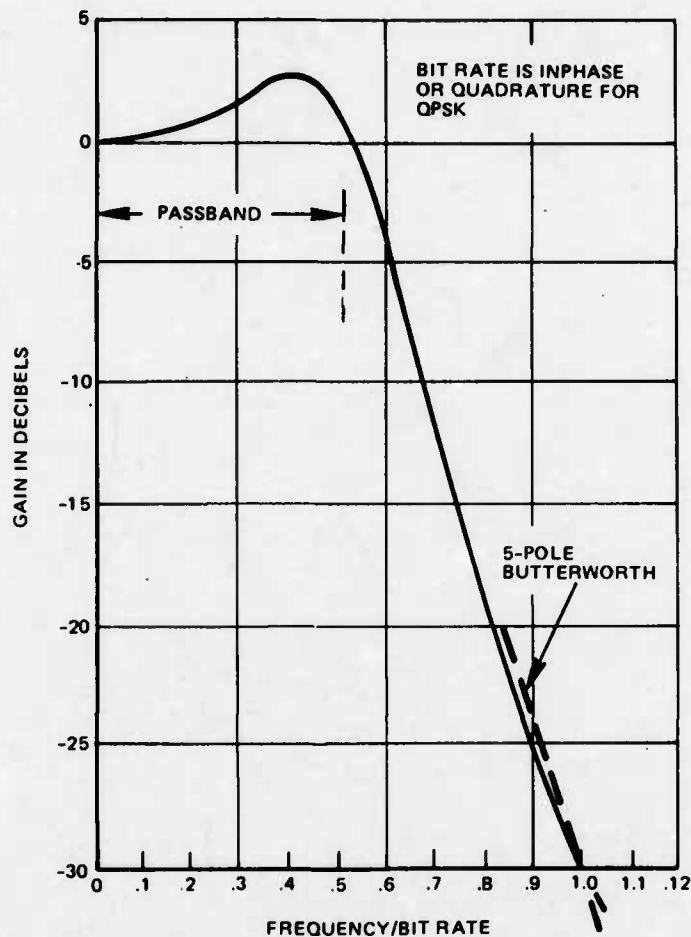
Figure 7-5. Equalized Filter for Receive

Identical results are obtained when the equalized filter placed at either the transmitter or the receiver. Clearly, the intersymbol interference is identical for either case. There is a reduction of noise power if the filter is at the receiver, and the factor by which the rms noise is reduced is computed from the response of the filter in cascade with the integrate-and-dump detector. In the alternate case where the filter is at the transmitter, the transmitter gain is increased to compensate for the power removed by filtering. Since the power spectrum of the QPSK signal is identical to the power response of the matched integrate-and-dump detector, the increase in transmitter gain is by exactly the same factor as previously computed for the reduction in rms noise in the receiver.

7.1.2.4   Staggered QPSK With Hard Limiting Transmitter

In a nonsaturating channel with symmetrical filters, QPSK and SQPSK have identical performance. If the transmitter hard limits after filtering, or if the channel hard limits, SQPSK maintains low out-of-band spectral emissions. The performance of SQPSK with hard limiting in the channel using the equalized filter of Figure 7-3 in both transmitter and receiver was obtained as a function of data rate. The results are plotted in Figure 7-6. Although degraded somewhat from the non-saturating case, the performance in a narrow bandpass is still good.

7.1.2.5   Response of Equalized Filters

The response of the equalized filters to a unit pulse is given in Figures 7-7 and 7-8. Figure 7-7 shows the output waveform of a single equalized filter while Figure 7-8 shows the output for the pulse passing through two cascaded equalized filters followed by an integrate-and-dump detector. Figure 7-9 shows the $\sin x/x$ impulse response of an ideal sharp cutoff filter. Note the similarity between Figures 7-8 and 7-9. This shows that use of a sampling detector following the equalized filters is equivalent to use of an integrate-and-dump detector.

PHASE DISTORTIONLESS FILTERS
EQUALIZED AMPLITUDE RESPONSE
RECEIVE AND TRANSMIT FILTERS
HARD LIMITING OF SQPSK, INTEGRATED AND DUMP
E_b/N_o = 10 dB

BIT RATE IS INPHASE OR
QUADRATURE
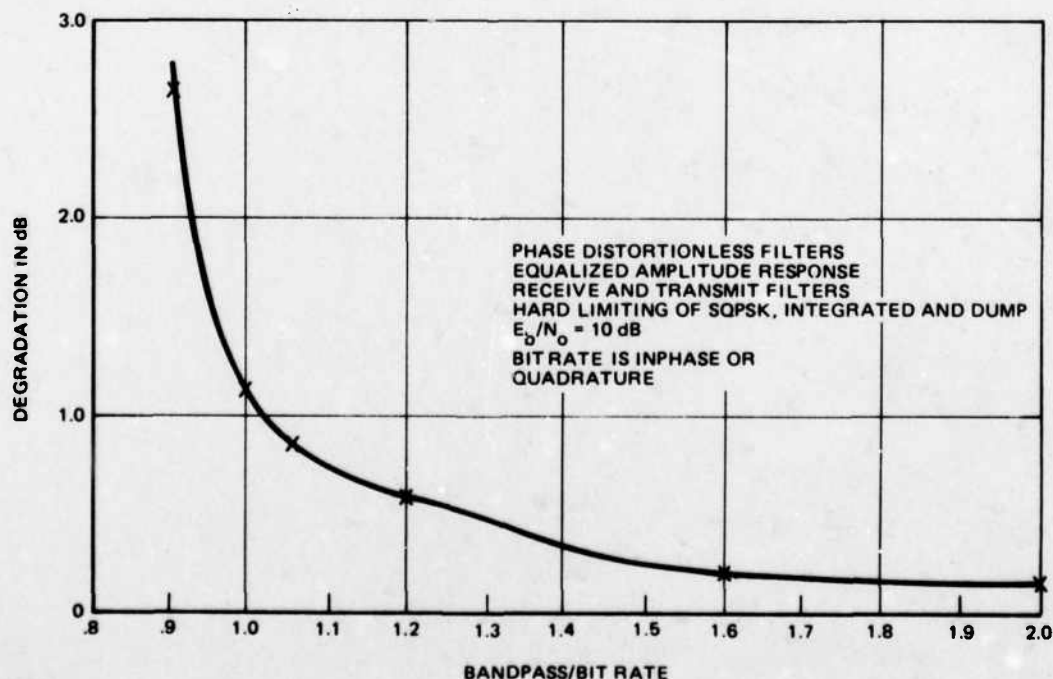
1176-6034
UNCLASSIFIED

Figure 7-6. Degradation for SQPSK With Hard Limiting

7-10

Figure 7-7. Pulse Response of a Single Equalized Filter

TIME

RESPONSE

0.00   0.20   0.40   0.60   0.80   1.00   1.20

0   1   2   3   4   5
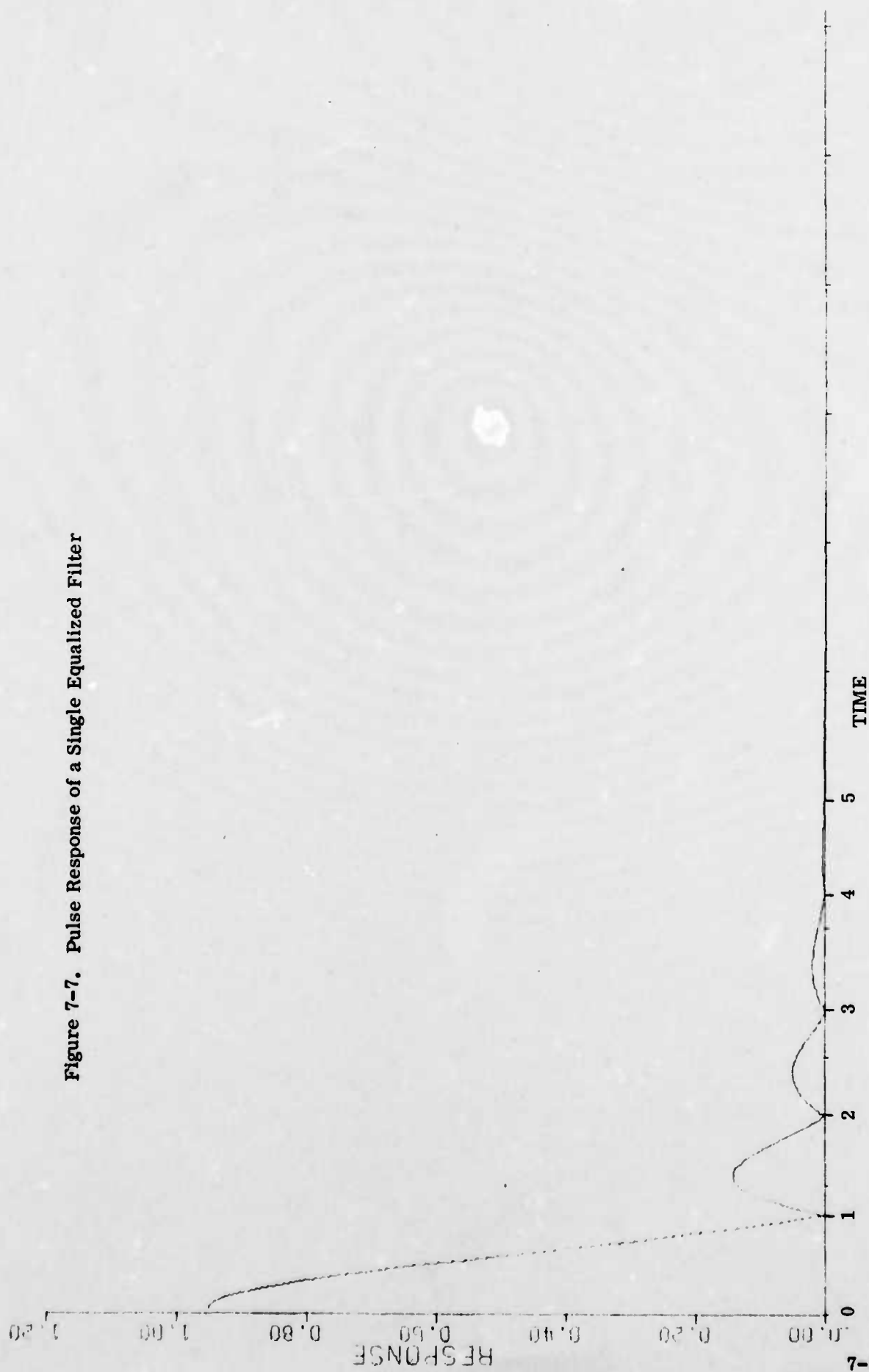
Figure 7-8. Pulse Response of Two Cascaded Equalized Filters Followed by Integrate-and-Dump Detector

7-12
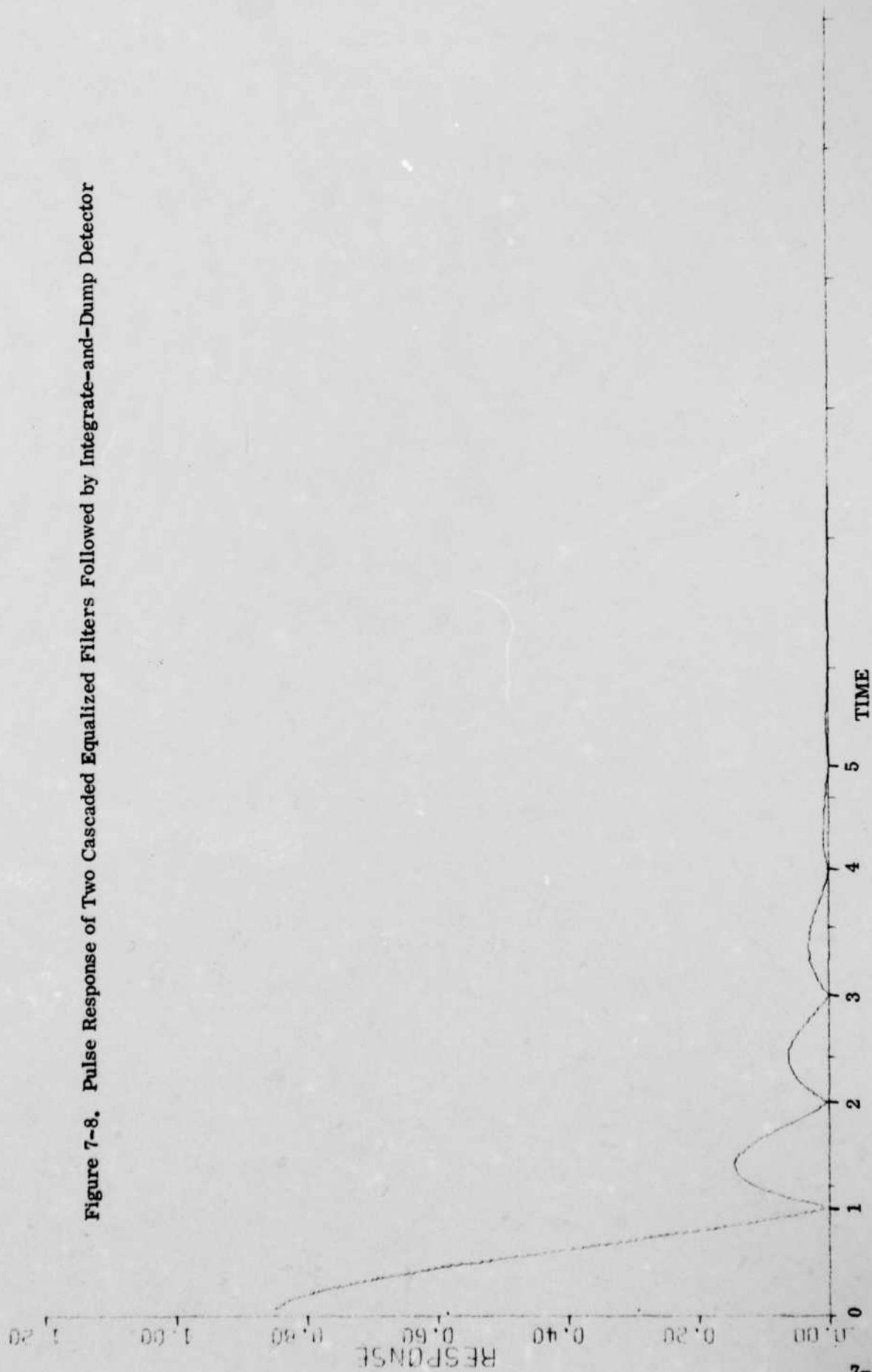
Figure 7-9. Sin(x)/x Impulse Response of Ideal Sharp Cutoff Filter
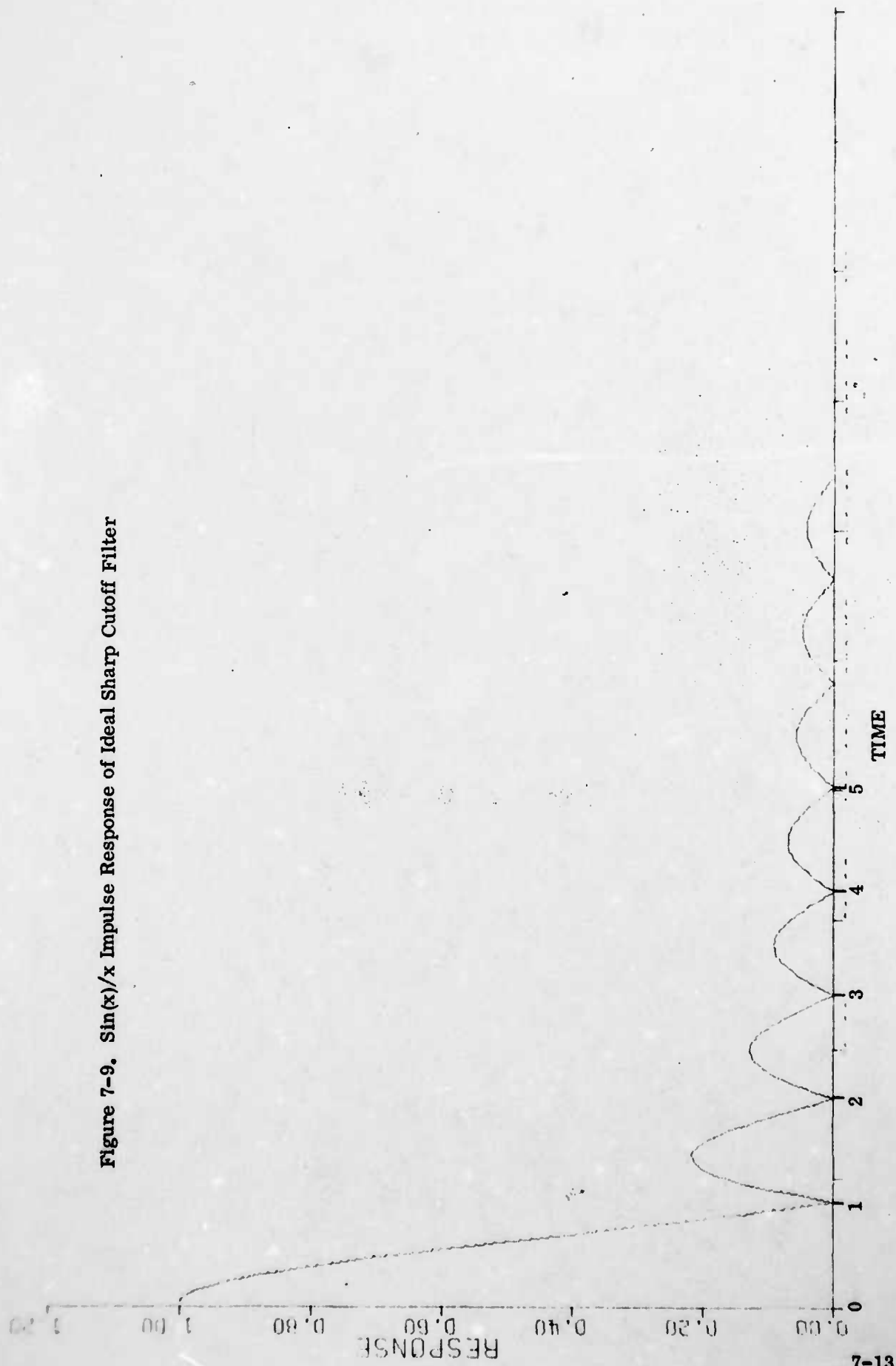
Figure 7-10 shows the frequency response of the two cascaded equalized filters and Figure 7-11 gives the overall frequency response of the cascaded filters followed by an integrate-and-dump detector. The resulting response of Figure 7-11 is very close to a maximally flat response, accounting for the performance improvement realized by use of the equalized filters.

7.1.3 COMPROMISE DESIGN FOR EQUALIZED SHARP CUTOFF FILTER

As shown in 7.1.2.2, the use of an equalized sharp cutoff filter is shown to improve QPSK modem performance in a narrow bandwidth by boosting the filter response near the edges of the passband. However, if the data rate is reduced from the design point, where the performance is almost as good as ideal QPSK, the degradation increases initially. The design point for the previous study sets the bandpass equal to the bit rate (inphase or quadrature for QPSK). The following computes performance for a compromise equalization, where the design point sets the bandpass slightly wider than the bit rate.

7.1.3.1 Design Point, Bandpass Equals 1.1 Bit Rate

The same computational procedure employed in 7.1.2.1 was repeated with the design point setting the bandpass equal to 1.1 times the bit rate, for both the transmit and the receive filters. Figure 7-12 plots the response of the equalized filter, where (in decibels)

$$G(\omega) = .5H(\omega) - 10 \log_{10}[1 + (\omega T_b/1.1\pi)^{10}] \qquad (4)$$

and

$$H(\omega) = 20 \log_{10}\left[\frac{1.065 - .256 \cos(\omega T_b) + .037 \cos(2\omega T_b) + .011 \cos(3\omega T_b)}{.857}\right] \qquad (5)$$

is the required equalization to be split between transmitter and receiver.

7.1.3.2 Design Point, Bandpass Equals 1.2 Bit Rate

Figure 7-13 is the response of the equalized filter with the design point setting the bandpass equal to 1.2 bit rate, for both the transmit and the receive filters. Now, (in decibels)

$$G(\omega) = .5H(\omega) - 10 \log_{10}[1 + (\omega T_b/1.2\pi)^{10}] \qquad (6)$$

Figure 7-10. Frequency Response of Two Cascaded Equalized Filters
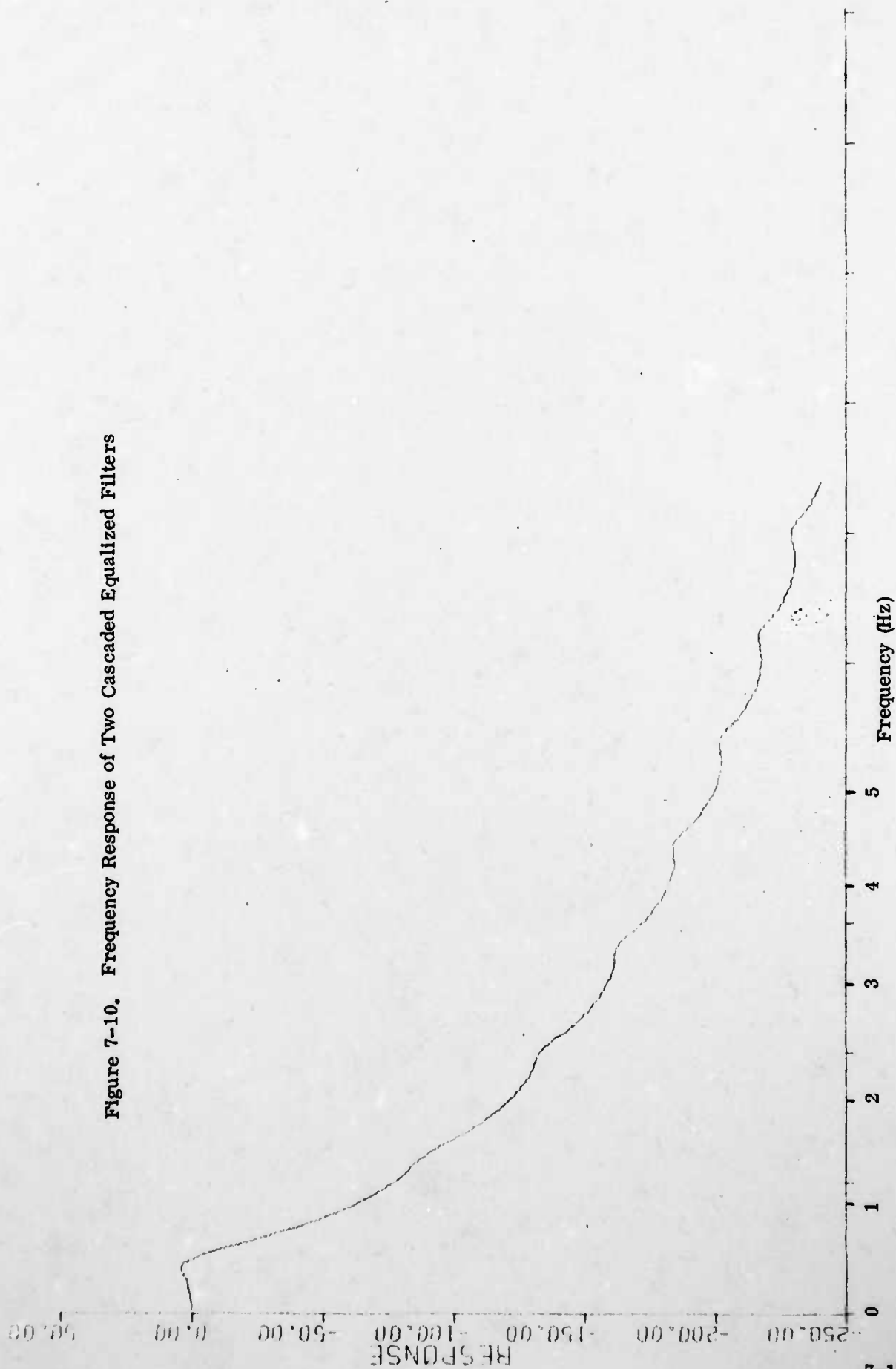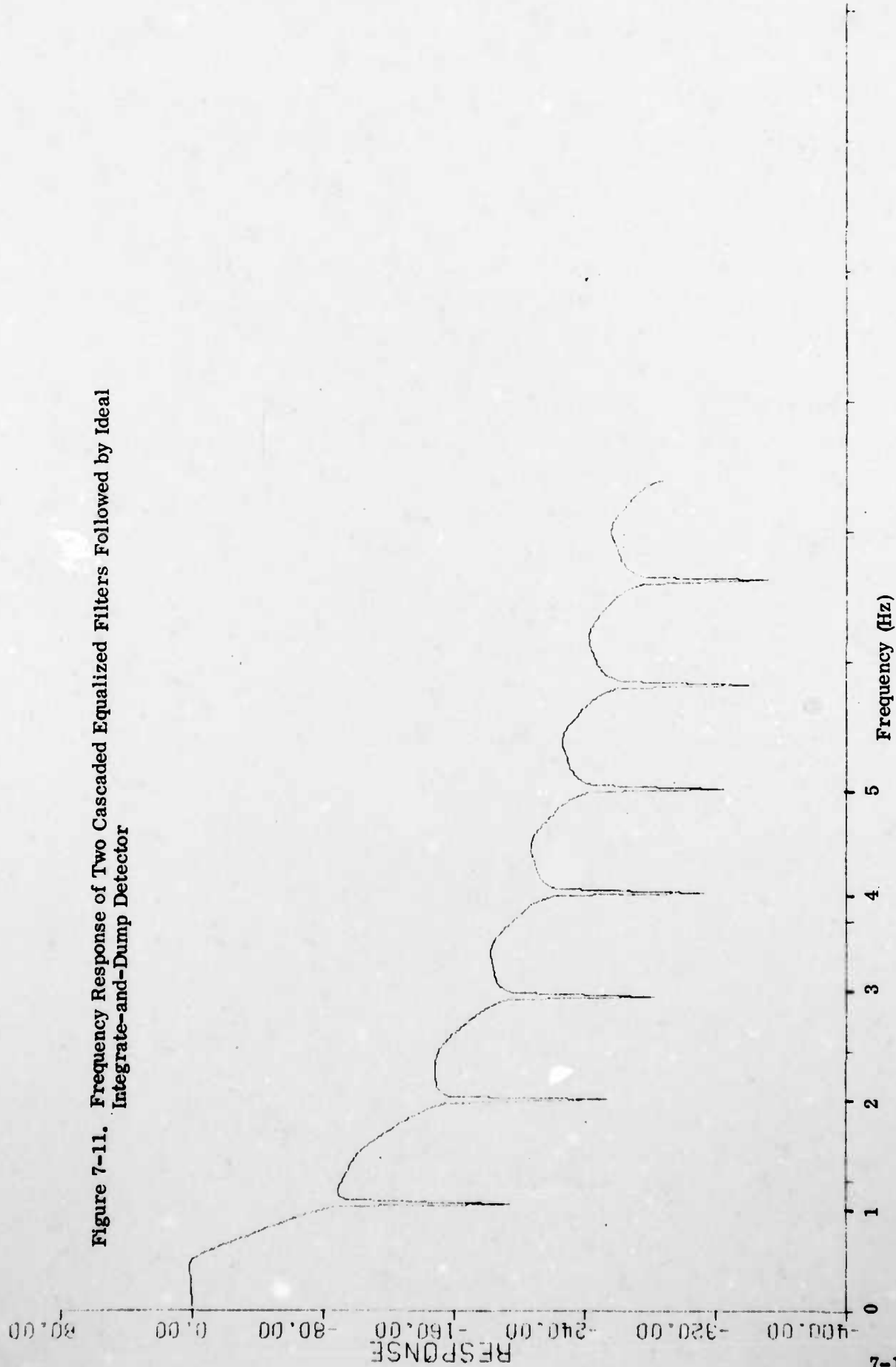
Figure 7-11. Frequency Response of Two Cascaded Equalized Filters Followed by Ideal Integrate-and-Dump Detector
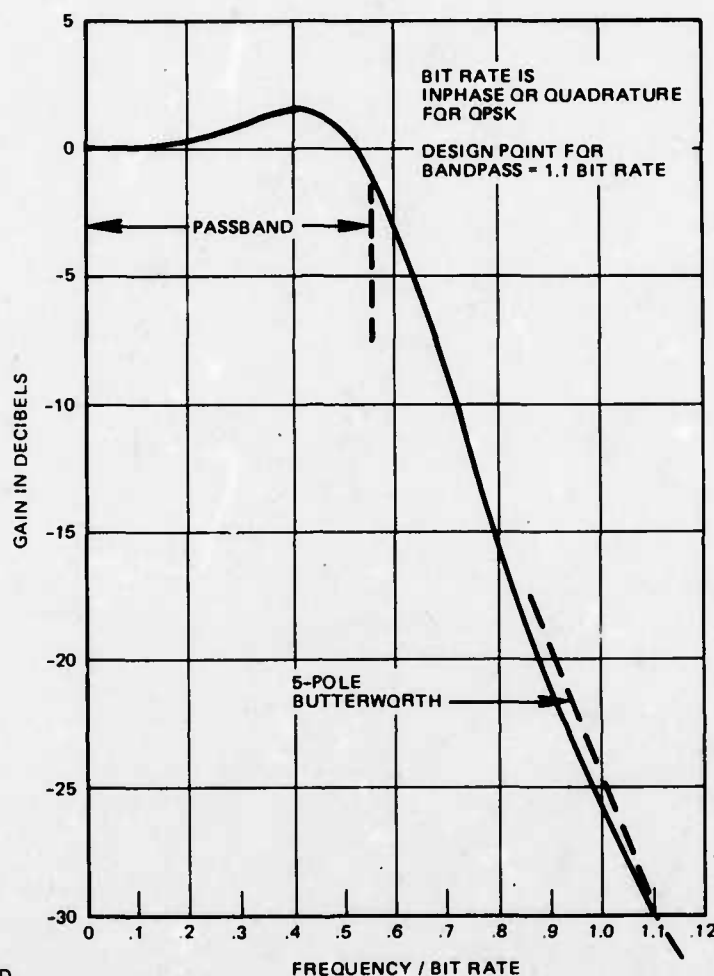
Figure 7-12. Equalized Filter for Transmit and Receive

and

$$H(\omega) = 20 \log_{10}\left[\frac{1.038 - .189 \cos(\omega T_b) - .003 \cos(2\omega T_b) + .027 \cos(3\omega T_b)}{.873}\right] \quad (7)$$

is the required equalization to be split between transmitter and receiver.

7.1.3.3   <u>Performance as Data Rate Varies From the Design Point</u>

The performance obtained for QPSK as the data rate is varied from the design point assumed for equalization is presented in Figure 7-14, for all three filters. The degradation is the decrease in E_b/N_o to achieve the same probability of error in an ideal QPSK system.

7-17

5

BIT RATE IS
INPHASE OR
QUADRATURE FOR QPSK

DESIGN POINT FOR
BANDPASS = 1.2 BIT RATE

0

PASSBAND

-5

GAIN IN DECIBELS

-10

-15

5-POLE
BUTTERWORTH

-20

-25

-30

0   .1   .2   .3   .4   .5   .6   .7   .8   .9   1.0 1.1 1.2 1.3 1.4

FREQUENCY/BIT RATE
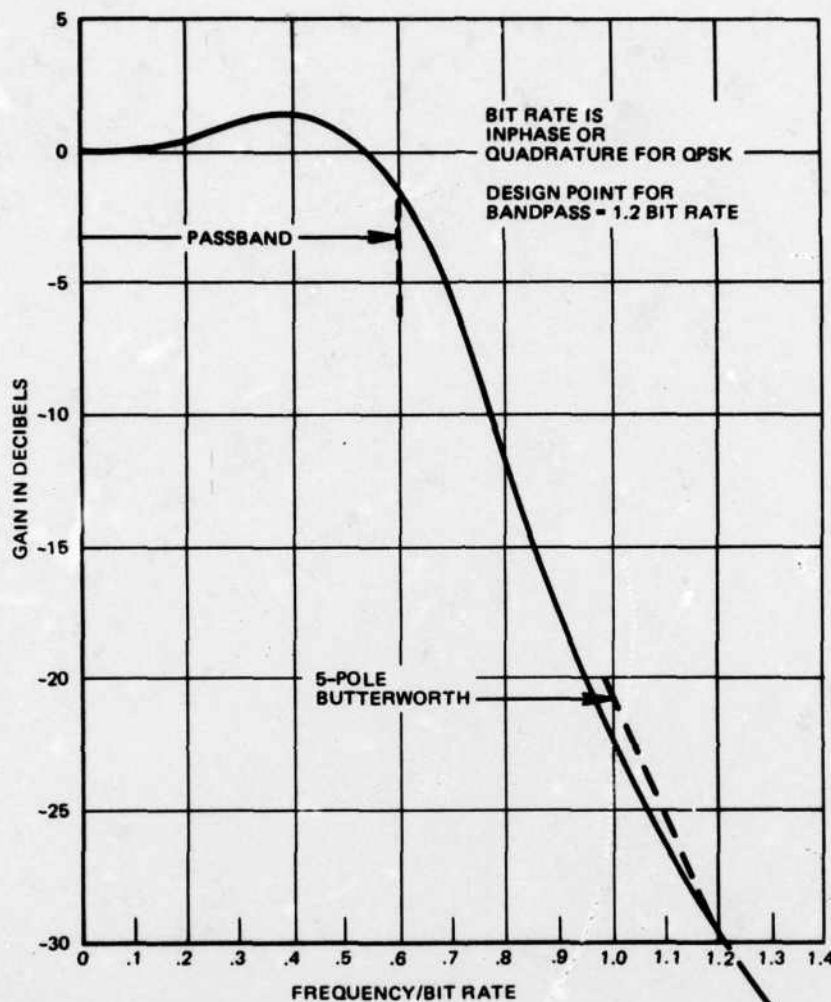
Figure 7-13.   Equalized Filter for Transmit and Receive

As can be seen, the degradation is close to zero for a data rate at the design point for each filter. The filter in Figure 7-12, for a design point with bandpass equal to 1.1 times the bit rate, is the suggested compromise since the degradation does not exceed approximately 0.5 dB for a bandpass equal to or exceeding the bit rate. Figure 7-13 provides a design where the degradation does not exceed 0.25 dB for a bandpass equal to or exceeding 1.1 times the bit rate.

For comparison, performance with phase distortionless Butterworth amplitude filters is plotted in Figure 7-15, as computed previously. Comparing Figure 7-15 with Figure 7-14, the benefit of equalization to boost the response near the edges of the passband is particularly evident for narrow bandwidths.
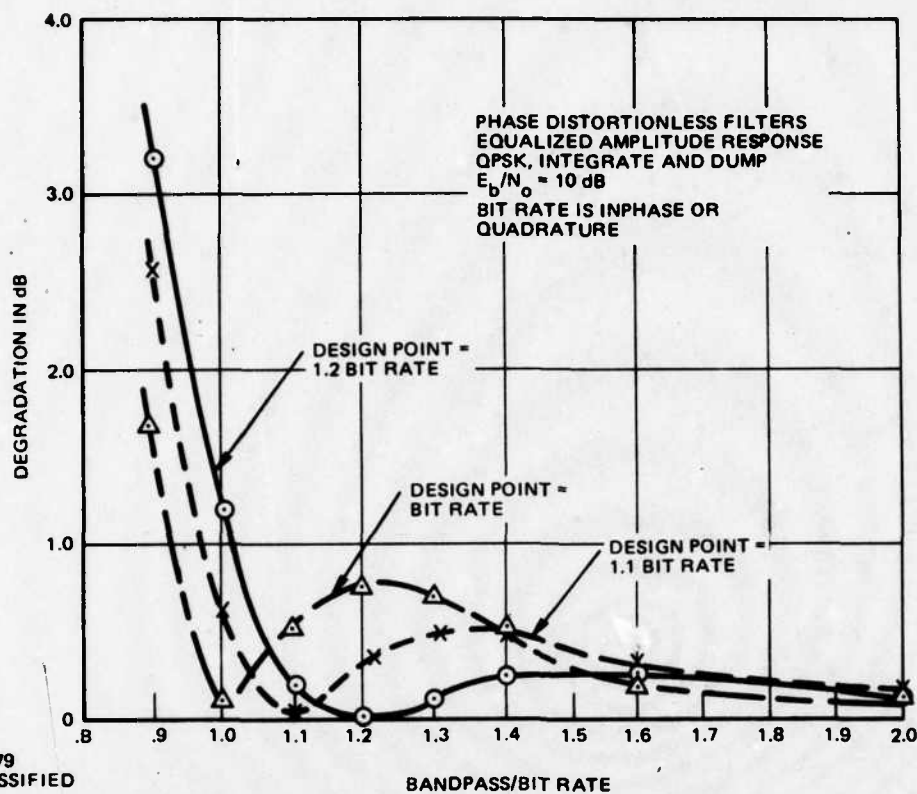
Figure 7-14. Degradation With Varying Data Rate in Fixed Filter in Transmitter and Receiver

7-19
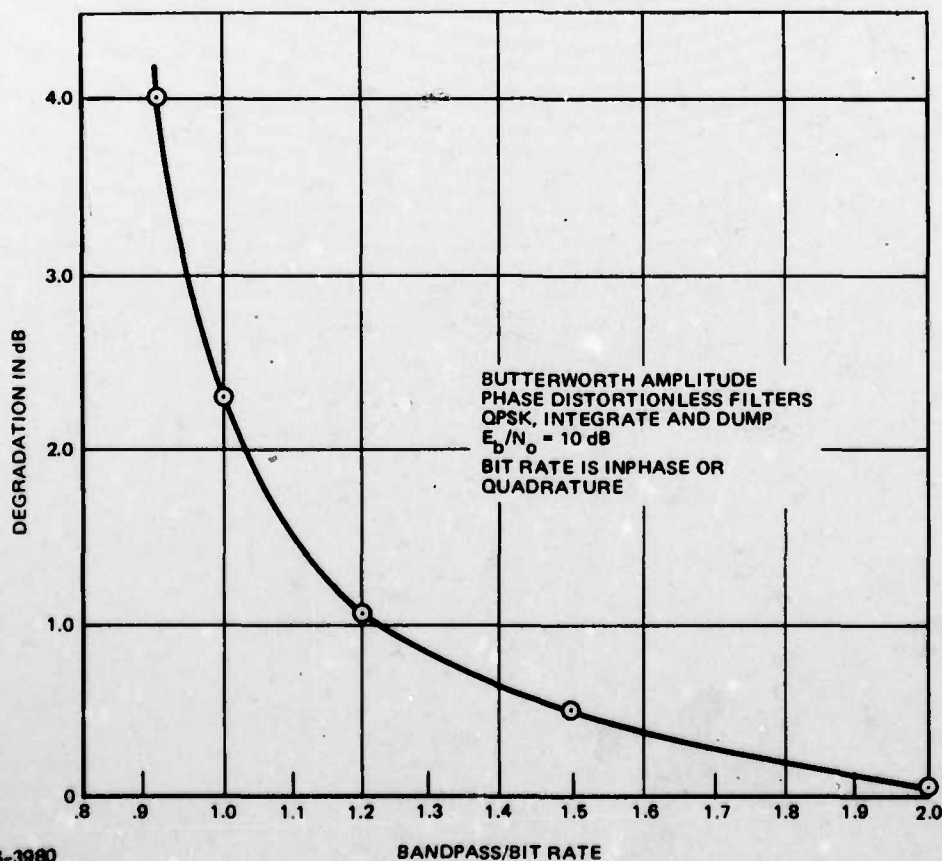
Figure 7-15. Degradation With Varying Data Rate in Fixed Filter in Transmitter and Receiver

7.2 UNCODED FDMA PERFORMANCE IN LINEAR (NONSATURATING) SQPSK CHANNELS

This section covers the considerations involved in the selection of channel filters to provide good bandwidth utilization in nonsaturating FDMA channels using PSK signaling. Performance curves are presented as a function of adjacent signal spacing for narrow bandwidth phase-distortionless Butterworth filters and for equalized filters as defined by Figure 7-3. Depending on the amplitude ratio of the adjacent signals to the center signal, equalized filters permit a signal separation of 1.1 to 1.25 times the bit rate with low degradation.

Simulations were performed to determine the system degradation resulting from adjacent signal interference and intersymbol interference as a function of the signal separation and filtering used. Five SQPSK signals are used with the outer signals equally spaced and symmetrically located about the center signal. Signal spacing FCENT refers to the ratio of the absolute value of the center frequency of the adjacent channels to the bit rate of the center channel data. Thus, if FCENT = 1., the adjacent channels are located at the first nulls of the center channel spectra.

Spacing FCENT was varied from 4 to 1 to cover the range of minimum system degradation to a level of interference that renders the system inoperable. A spacing of FCENT = 4 has essentially zero spectral overlap between adjacent signals and system degradation is determined solely by the intersymbol interference contributed by the filters. In all cases, the adjacent channels have power 6 dB above that of the center signal for which the probability of bit error is computed. This power differential represents a worst case expected value for the ground mobile forces application. For lesser power ratios, the degradation would decrease for close spacings of signals but would remain the same for wide signal separation.

Two input signal-to-noise ratios were used. $E_b/N_o = 10$ dB assumes a noise level low enough that adjacent channel and intersymbol interference tends to dominate while $E_b/N_o = 2$ dB assumes a noise level which dominates for lower values of adjacent channel interference. The latter can be used to estimate degradation with a rate-1/2 error correcting code. Degradation is measured in the center channel.

Preliminary simulations showed the necessity for having filters at both the transmitter and receiver, especially when the accessing signals have unequal

power. For example, if the adjacent channels are each 6 dB stronger than the center channel, the peak power level of the second lobe of either adjacent signal is only 7 dB below the maximum level of the center signal. For close equal spacing, the ratio of desired signal to total adjacent channel interference can drop to less than 4 dB. With no channel filtering, this spectral overlap produces severe degradation. With only transmit filtering, the out-of-band energy in the receiver produces high degradation. With only receiver filtering, no attenuation is provided for the adjacent channel spectral energy falling into the receiver passband. The problem becomes even more severe when nonlinearities are introduced into the channel, increasing the magnitude and extent of the interfering spectra. When both transmit and receiver filtering is employed, the combination suppresses in-band and out-of-band interference sufficiently to produce minor degradation until a spacing at which the filter 3-dB points approach each other.

Figures 7-16 and 7-17 show the system degradation as a function of signal spacing for the Butterworth and the equalized filters in the transmitter and receiver. Figure 7-16 shows the degradation at a received E_b/N_o = 2 dB, where coding would be used. Figure 7-17 shows the corresponding degradation at E_b/N_o = 10 dB for an uncoded channel.

776-3981
UNCLASSIFIED
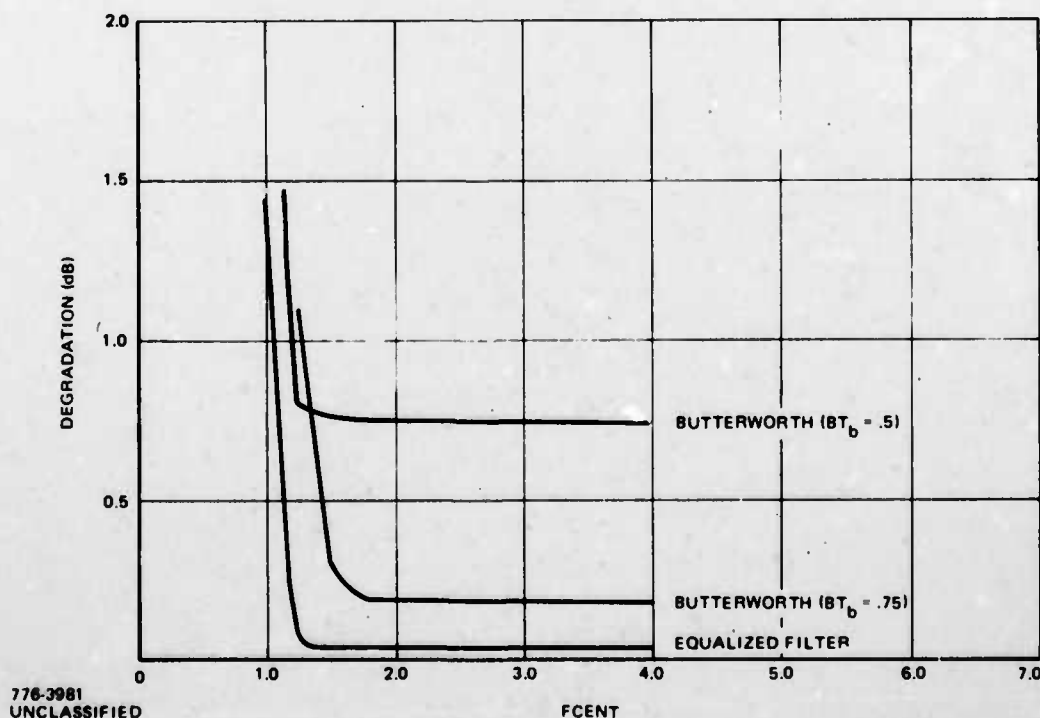
Figure 7-16. System Degradation as a Function of Signal Spacing at E_b/N_o = 2 dB. (Linear Channel, Adjacent Signals +6 dB)

7-21
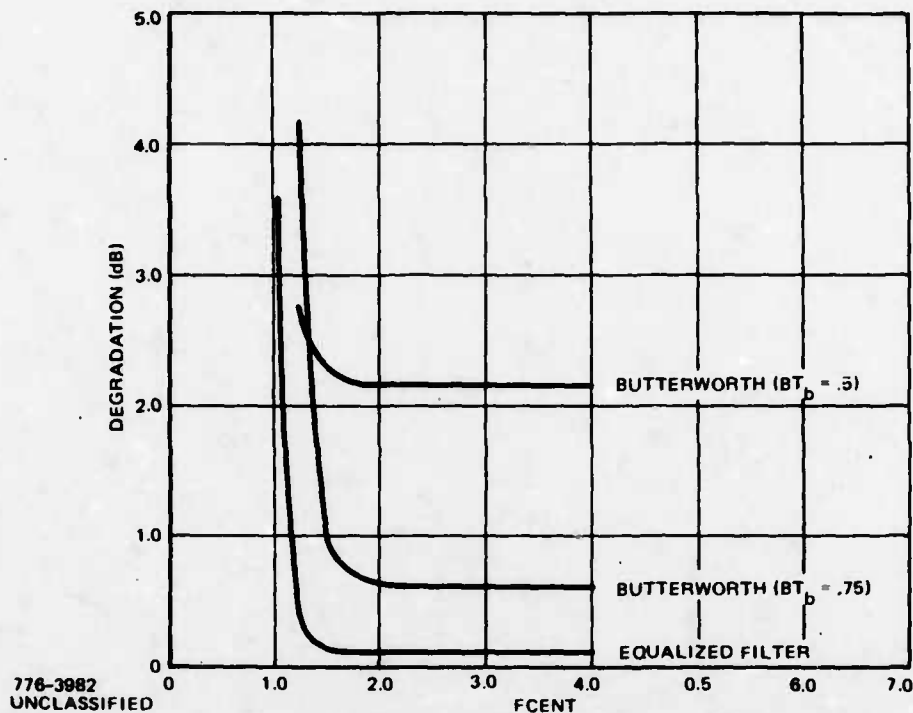
Figure 7-17. System Degradation as a Function of Signal Spacing at E_b/N_o = 10 dB. (Linear Channel, Adjacent Signals +6 dB)

As shown in these figures, the degradation at large spacings converges to that value due solely to the intersymbol interference produced by the system filters. The least intersymbol interference results from use of the equalized filters. The worst intersymbol interference is produced by the 5-pole Butterworth filter with $BT_b = 0.5$. As the frequency spacing is reduced, the degradation has a "knee" corresponding to values of spacing at which the spectral overlap of the adjacent channels overwhelms the intersymbol interference and system noise levels.

Below this critical spacing, system degradation increases very rapidly. As shown, the least degradation occurs for all spacings with the equalized filter. Similarly, the closest signal spacing is obtained for the equalized filter. Depending on the allowable degradation and E_b/N_o at which the system operates, these curves present the frequency separation possible with each filter type. As an example, for E_b/N_o = 2 dB and an allowable system degradation of 0.5 dB, the following separations result:

$$FCENT = 1.4 \text{ for Butterworth } (BT_b = .75)$$
$$FCENT = NA \text{ for Butterworth } (BT_b = .5)$$
$$FCENT = 1.1 \text{ for equalized filter}$$

Similarly, the corresponding degradations at any spacing can be obtained. For example, at $E_b/N_o = 10$ dB and FCENT = 1.25, the following degradations result:

$$\text{Degradation} = 4.17 \text{ dB for Butterworth } (BT_b = .75)$$
$$\text{Degradation} = 2.66 \text{ dB for Butterworth } (BT_b = 0.5)$$
$$\text{Degradation} = 0.36 \text{ dB for equalized filter}$$

Use of the equalized filters permits a signal separation FCENT in the range of 1.1 to 1.25 with very small degradation. Employing QPSK or SQPSK, this would permit placing 360 to 410 full duplex, rate-1/2 coded signals in the allowable 45 MHz bandwidth of the ground mobile forces channel (assuming sufficient satellite channel backoff to have a linear channel). An increase in capacity by approximately 25 percent is realized with the equalized filter, compared with the Butterworth characteristic $(BT_b = .75)$.

7.3 UNCODED FDMA PERFORMANCE IN SATURATING SQPSK CHANNELS

This section evaluates a variety of cases of FDMA performance versus signal spacing for nonlinear PSK channels. The channel modelled is shown in Figure 7-18. Saturating elements include hard limiters, the Phase II TWT, and the Hughes' TWT. The models used are discussed in 3.3. Both the phase distortionless 5-pole Butterworth $(BT_b = .75)$ and the equalized filter (Figure 7-3) are considered, with identical filters in the transmitter and receiver. Both the desired case of five equal level signals and the worst case of the adjacent signals +6 dB above the center signal are evaluated. Results show that the use of equalized filters permit closer signal spacing for a given level of degradation than occurs when Butterworth filters are used, except in the case of hard limited transmitters. In general, with TWT amplifiers and equalized filters, the signals can be spaced by 1.2 times the bit rate with degradation comparable to that at large spacing. An optimum backoff drive exists for each TWT which minimizes the total system loss for a given signal spacing.
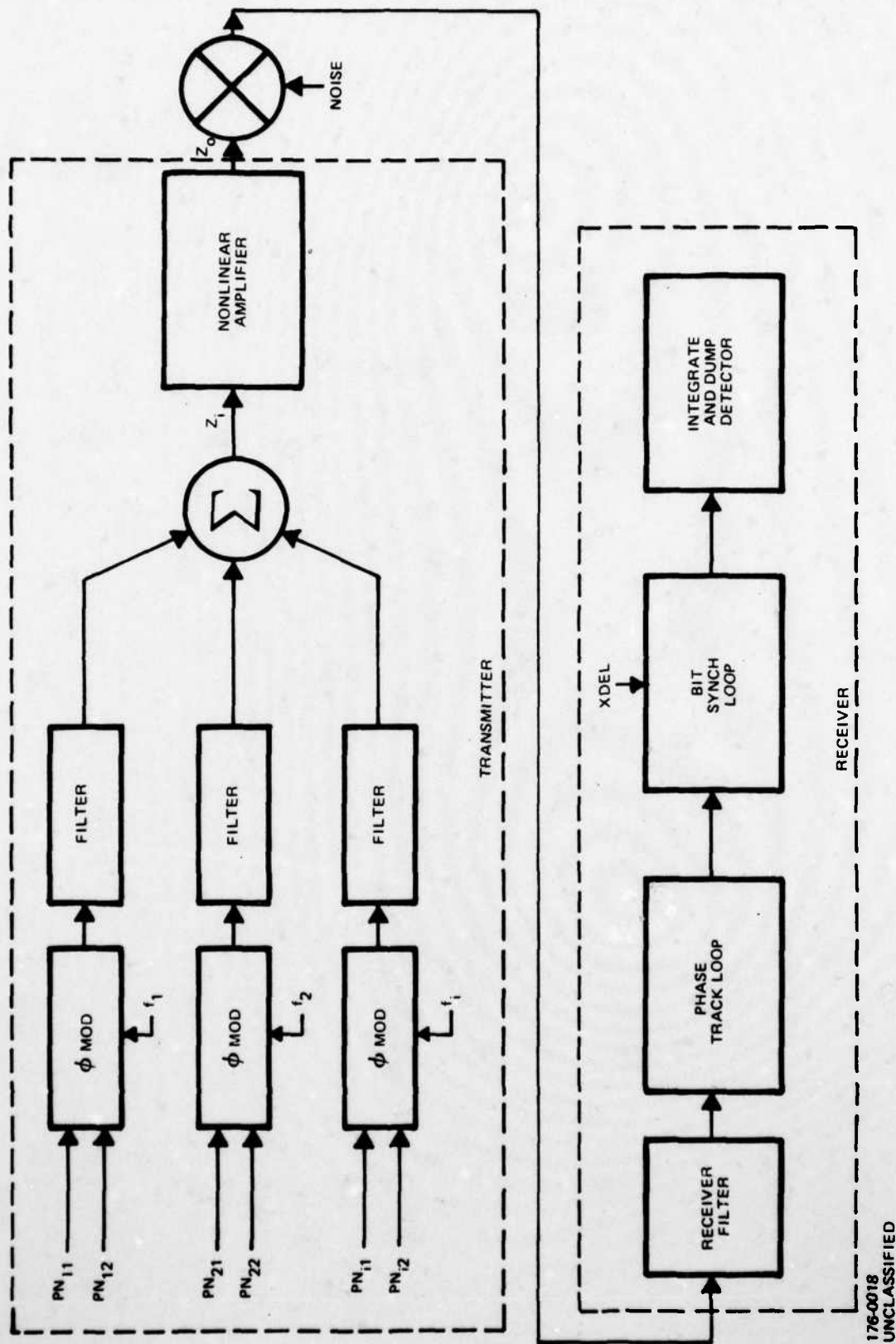
Figure 7-18. Block Diagram of System Simulated

7.3.1 SIMULATION CONSIDERATIONS

In all cases, five SQPSK signals are used with equal frequency spacing between adjacent signals. Five equally spaced signals generate third-order intermodulation that falls directly on the center channel with a level independent of the level (or presence) of the center signal. This could also be generated using only two equally-spaced signals on one side of the center signal; however, the five signal case more accurately emulates the ground mobile forces scenario. Parameter FCENT determines the signal spacing and is given by the ratio of the absolute value of the frequency separation to the bit rate of the center channel data. Thus, if FCENT = 1, the two inner adjacent channels are located at the first nulls of the center channel spectra and the two outer channels are symmetrically placed at twice this frequency spacing about the center channel. The bit transition times are staggered by one-eighth of the bit time for each of the five SQPSK signals and independent PN sequences are used for the signals to minimize cross-correlation effects in the channel.

Spacing FCENT is varied over the range of 2.0 to 1.0 (or wherever degradation becomes severe). Degradation remains relatively constant for FCENT > 2.0 at the value produced by intermodulation and intersymbol interference, since these sources of degradation overwhelm any effects of adjacent signal spectral overlap. All curves are plotted for received E_b/N_o = 2 dB, where coding would be used, or E_b/N_o = 10 dB, corresponding to an uncoded system.

System degradation is measured for the center channel by first determining the receiver input signal-to-noise ratio (E_b/N_o) which would produce the same resulting probability of bit error in an ideal system, converting this E_b/N_o to decibels, and subtracting this number from the actual receiver input E_b/N_o used.

Automatic gain control (AGC) is used to normalize the total received signal energy to unity at the receiver input. Next, this normalized energy is boosted by the factor β to bring the energy of the center channel to unity to relate the user-specified E_b/N_o to the central signal (for which the probability of bit error is computed). If the composite waveform contains N adjacent signals, with the kth adjacent signal having power P_K and the center channel having power P_C, then

$$\beta = \frac{P_C + \sum_{k=1}^{N} P_K}{P_C} \tag{8}$$

As an example, for 5 accessing signals with the four adjacent signals each +6 dB above the center signal (for which degradation is determined) N = 4, P_C = 1, and P_K = 4 (6 dB) for k = 1, 2, 3, 4. Therefore, β = 17 since the center signal contributes one seventeenth of the total signal power. While this ratio is strictly accurate only when the TWT is operating at large backoff, the suppression effects for low backoff are not large and are therefore not compensated in the normalization process.

In order to ascertain the degradation due to various amounts of AM-PM conversion, the linear phase model of Figure 3-11 was used. Five equal level signals with spacing FCENT = 4 accessed the channel and both the transmitter and receiver contained equalized filters. Figure 7-19 is plotted for an average input power P_A 10 dB above the power level P_F where the linear phase shift with dB begins. At this drive level, the degradation due to AM-PM conversion is well into the plateau region of maximum degradation. For P_A less than about 5 dB (above P_F) the degradation decreases approximately linearly until a value of $P_A \leq -17$ dB below which degradation remains at that value of degradation produced by the filters.

776-39B3
UNCLASSIFIED
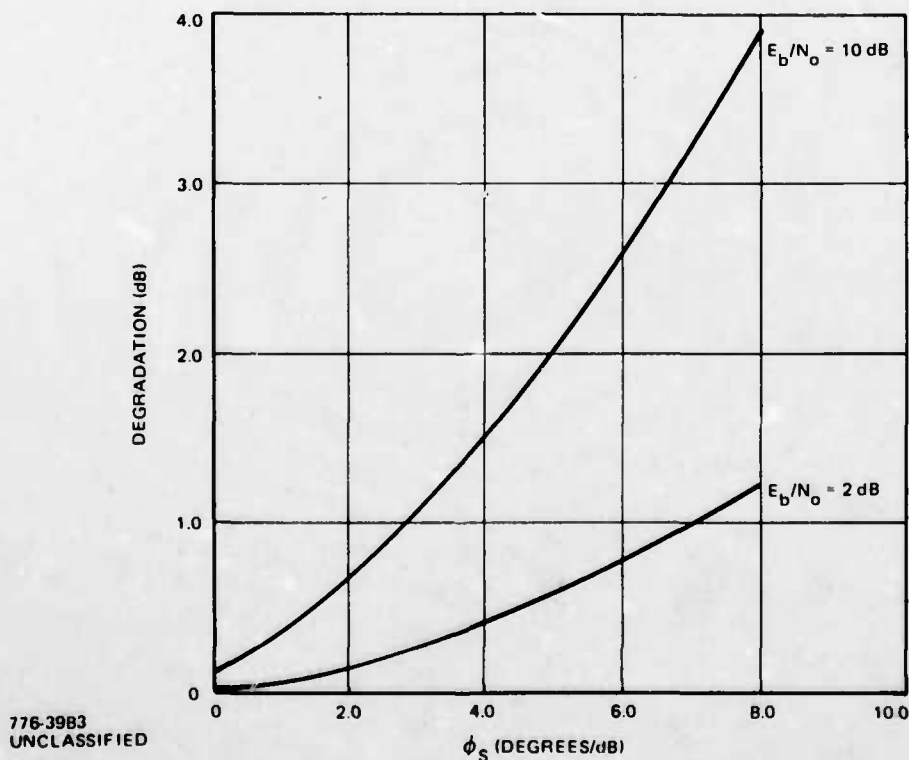
Figure 7-19. Degradation as a Function of Linear AM-PM Conversion for Five Equal Signals (Equalized Filters in Transmitter and Receiver)

MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A
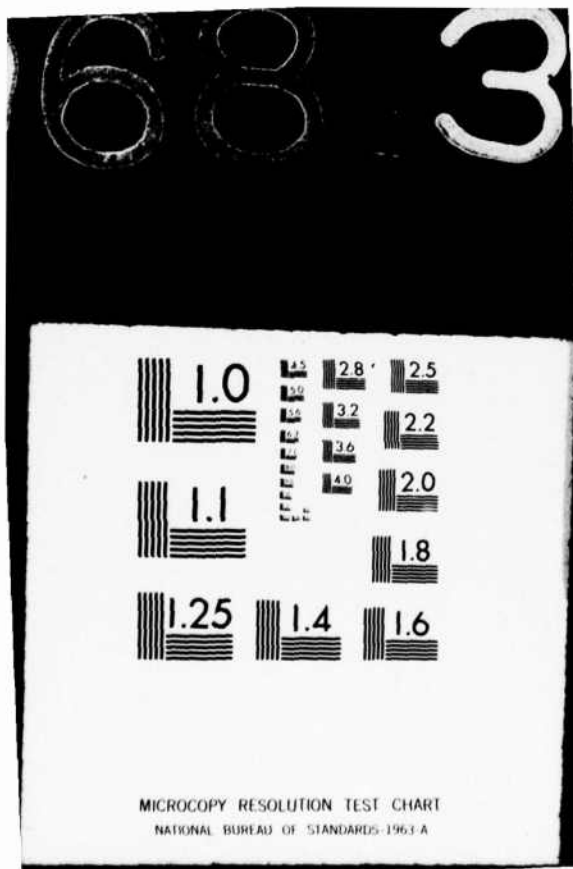
7.3.2   PERFORMANCE WITH PHASE II TWT: EQUAL SIGNAL LEVELS

The Phase II TWT is used in the satellite, simultaneously accessed by five equal level SQPSK signals. The power differential of 0 dB represents the desired condition of operation for the ground mobile forces application. Note that all backoffs refer to average output power backoff. The peak power for the cases run is 7.0 dB above this level so that with the TWT model used, the signal peaks are clipped in amplitude for backoffs less than 7.0 dB. Similarly, the signal peaks receive a phase shift of 36° for backoffs less than 7.0 dB.

Figure 7-20 shows the spectrum of the signals in a linear channel with equalized filters and the center signal absent. Figures 7-21 and 7-22 portray the intermodulation produced by the Phase II TWT at a backoff of 4 dB. Figure 7-21 shows the spectrum at the output of the TWT while Figure 7-22 shows the corresponding spectrum following the receiver filter. Since the center signal is absent and the frequency spacing is so large that no significant fundamental spectral overlap occurs, the spectrum shown results from intermodulation distortion. As shown, the peak signal to intermodulation ratio in the center channel is about 12 dB (since the center channel power is 6 dB less than that of the adjacent channels). Alternate intermod analyses are given in Appendix B.

Figures 7-23 through 7-30 show the degradation produced by the Phase II TWT as a function of output power backoff and signal spacing. Figures 7-23 and 7-24 assume use of the 5-pole zero phase Butterworth filters (BT_b = .75) at E_b/N_o = 2 dB and 10 dB. Figures 7-25, 7-26, 7-29, and 7-30 assume use of the equalized filters (Figure 7-3) at E_b/N_o = 2 dB and 10 dB. Figures 7-27 and 7-28 assume no system filters.

Figures 7-25 and 7-26 show a decrease in degradation for FCENT around 1.3. This dip does not occur in Figures 7-23 and 7-24. This effect appears due to the equalized filter peak and its interaction with the adjacent signal inter-modulation spectrum. As shown in Figure 7-21, the saturating TWT produces adjacent spectral peaks down only 12 dB from the main peak (in the linear channel the ratio of these peaks is 28 dB). At a spacing FCENT $\doteq$ 1.5, this adjacent peak is accentuated by the receiver filter peak. At a spacing FCENT $\doteq$ 1.3, the receive filter peak falls over the adjacent signal spectral null. Since the Butterworth filter has flat response in its passband, there is no comparable peaking effect.
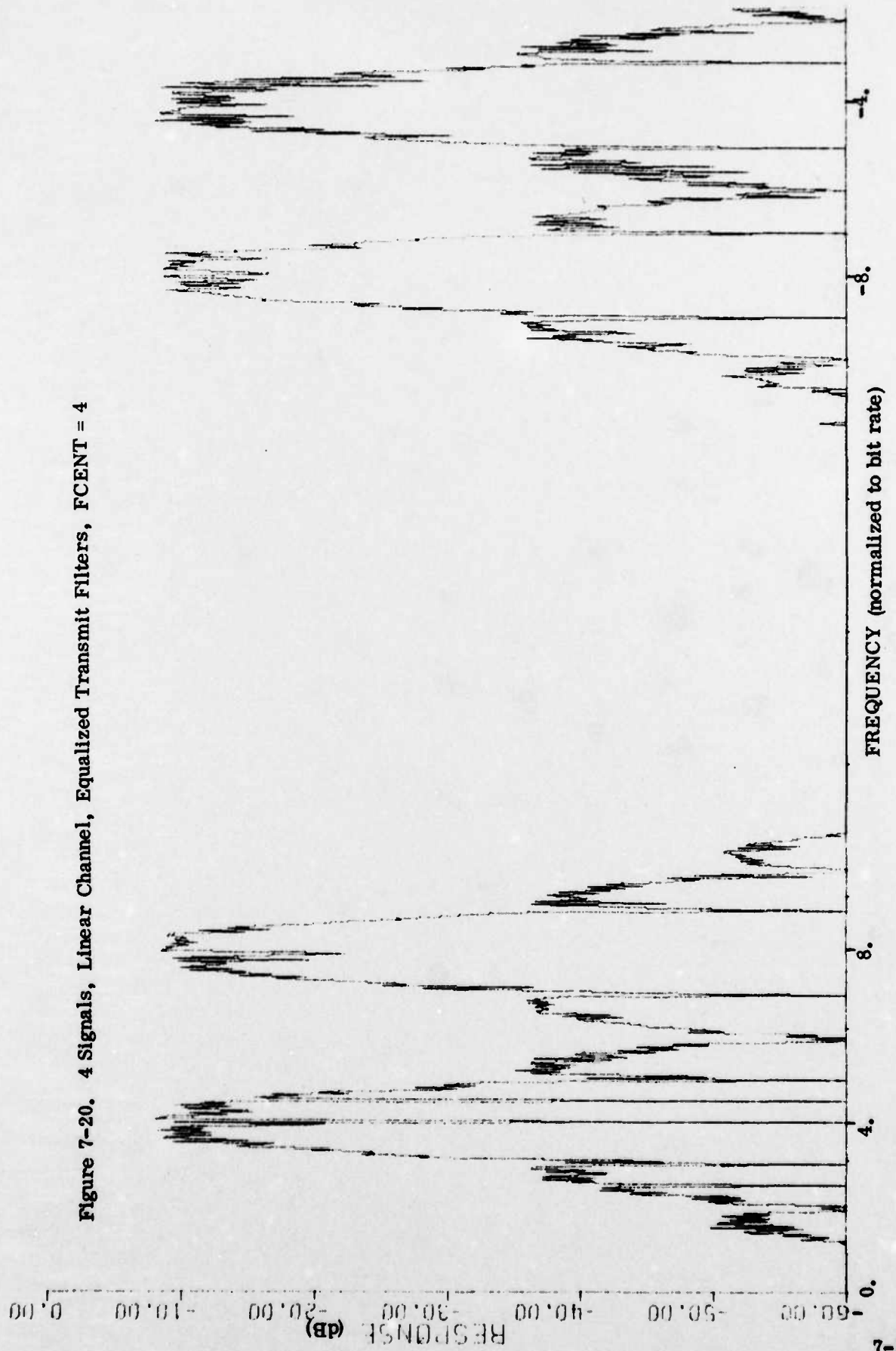
Figure 7-20. 4 Signals, Linear Channel, Equalized Transmit Filters, FCENT = 4
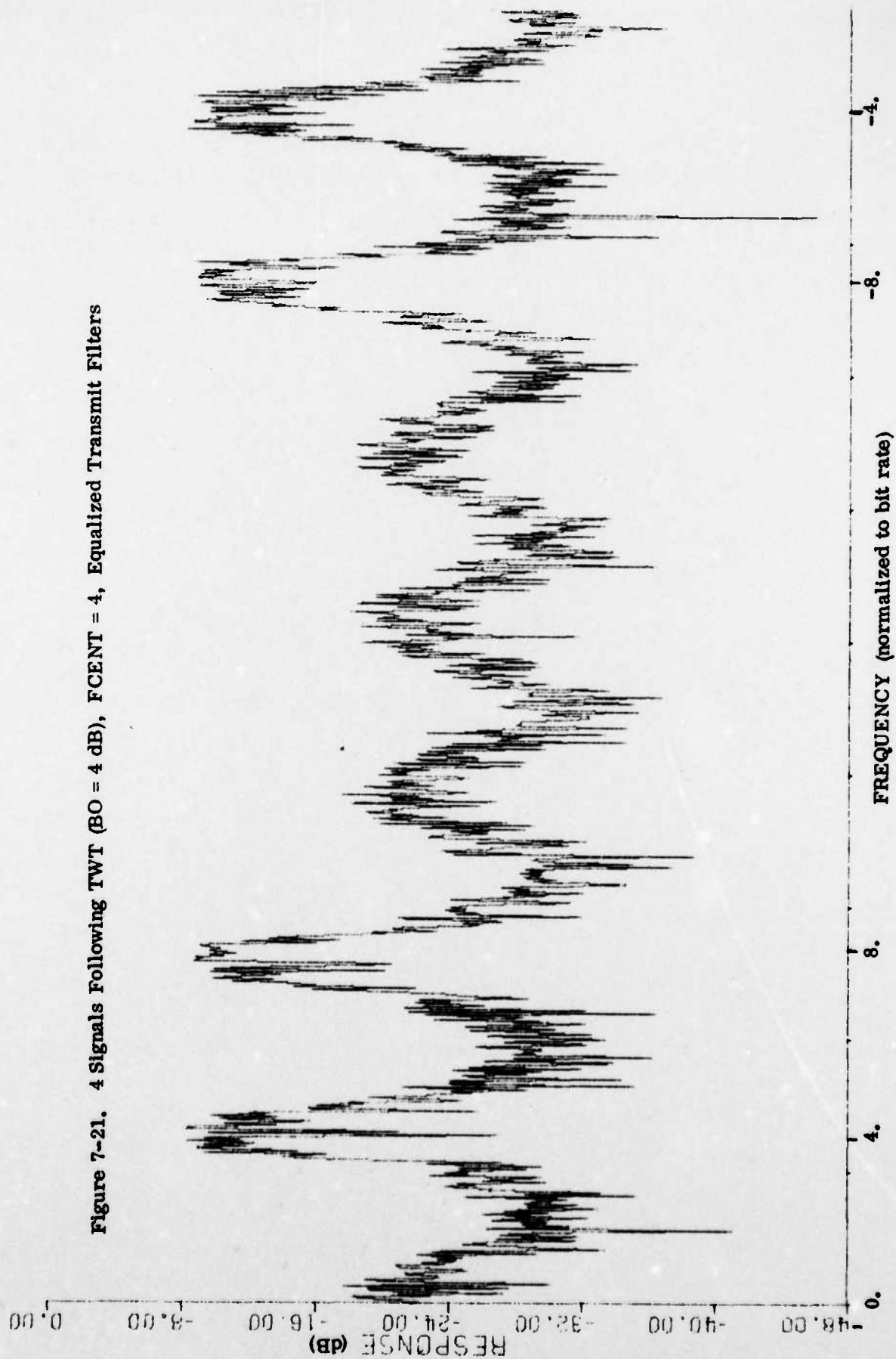
Figure 7-21. 4 Signals Following TWT (BO = 4 dB), FCENT = 4, Equalized Transmit Filters

Figure 7-22. 4 Signals Following Receive Filter, TWT BO = 4 dB, FCENT = 4, Equalized Receive and Transmit Filters
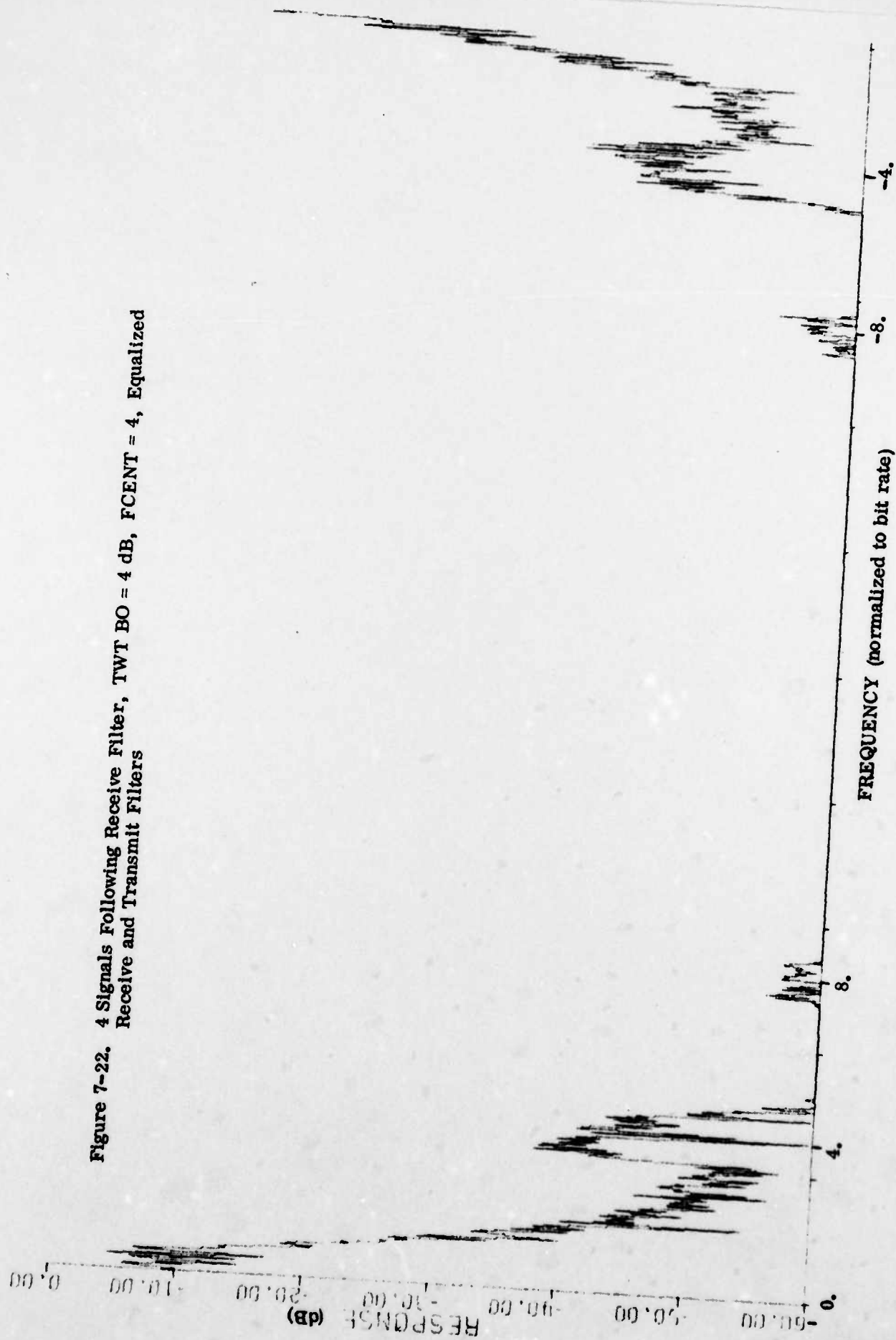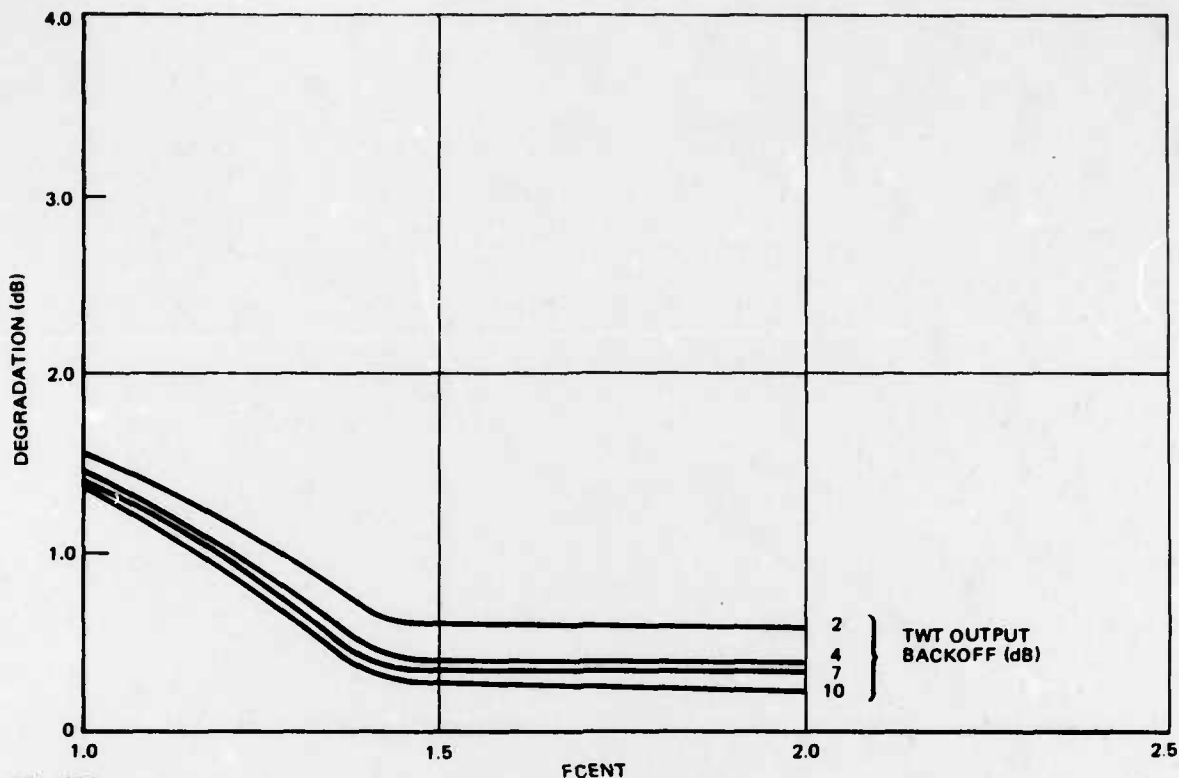
Figure 7-23. Degradation With Butterworth Filters (BT_b = .75) at E_b/N_o = 2 dB

Figures 7-27 and 7-28 show the degradation occuring when no filtering is used in the transmitter or receiver. As shown, degradation is significantly higher than when filtering is used, especially for $1.0 < FCENT < 2.0$, the separation of importance in this study. This is due to the increased intermodulation due to the wider signal spectra through the TWT as well as the large out-of-band energy in the receiver. This degradation would increase rapidly if the adjacent signal levels were increased above that of the center signal. Thus, as noted earlier, both transmit and receiver filtering should be used.

Figure 7-29 gives the E_b/N_o required to achieve a probability of bit error = 10^{-5} as a function of signal spacing and TWT output backoff. Equalized filters are used in the transmitter and receiver. The reason for replotting Figure 7-26 for a fixed error rate is that the curves for fixed E_b/N_o predict degradation accurately only for lower values of degradation. At higher degradation, Figure 7-26 is a lower bound while Figure 7-29 gives the true E_b/N_o required. In the ideal channel, an $E_b/N_o \cong 9.6$ dB achieves this error rate. As TWT backoff is decreased, the increased intermodulation pushes the E_b/N_o required higher. Almost a four dB increase in

7-31

E_b/N_o is required to achieve a 10^{-5} error rate when output backoff = 2 dB, even for large signal spacing. The intermodulation places a floor on achievable probability of error so that the required E_b/N_o to achieve very low error rates becomes infinite for low backoff.
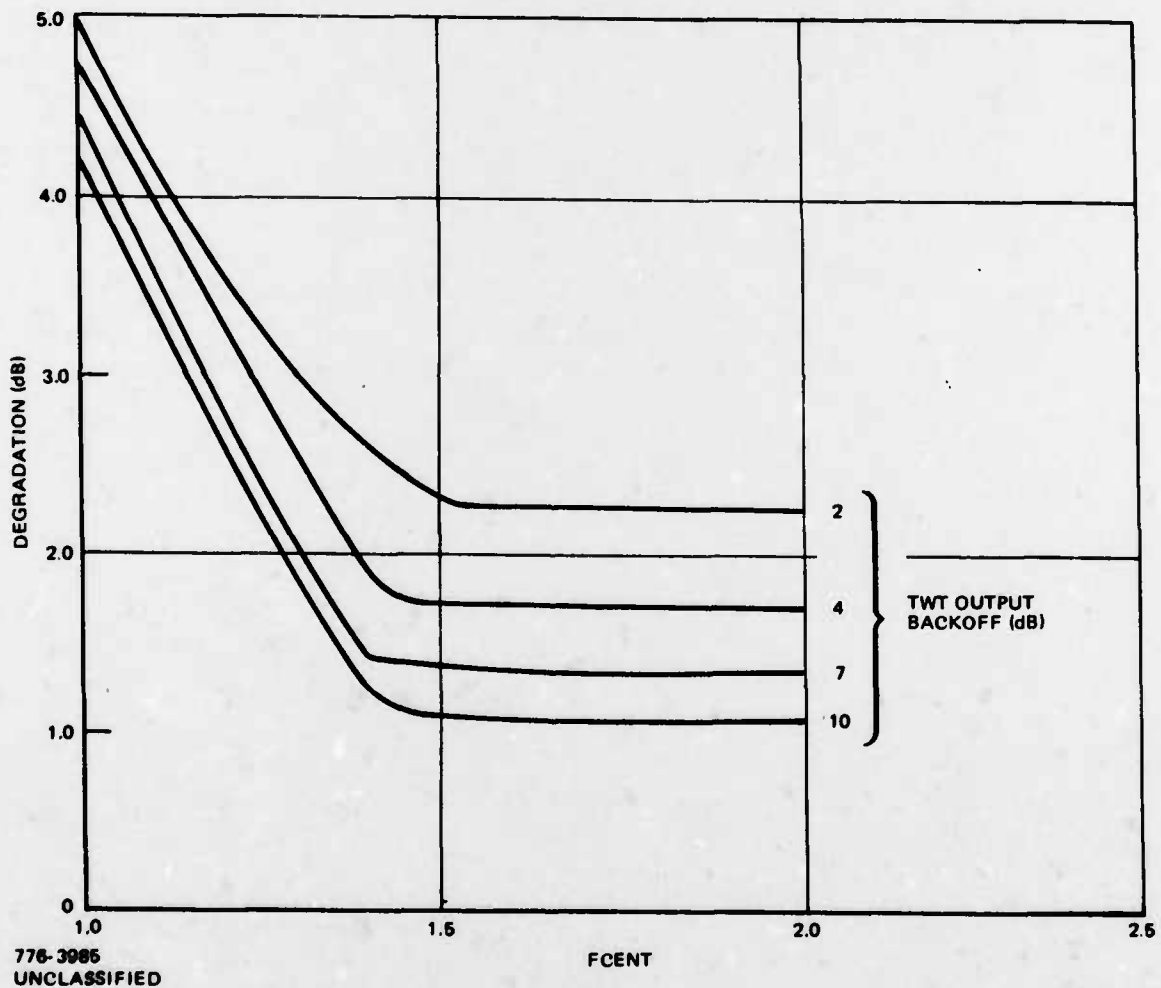
Figure 7-24. Degradation With Butterworth Filters (BT_b = .75) at E_b/N_o = 10 dB

776-3986
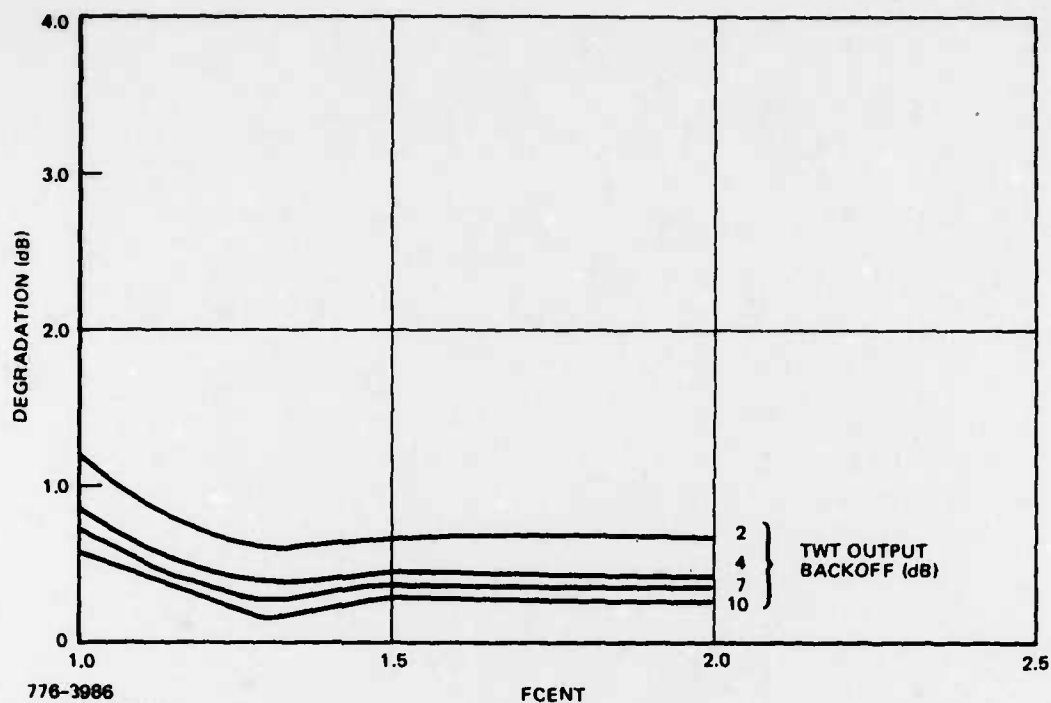
Figure 7-25. Degradation With Equalized Filters at E_b/N_o = 2 dB

776-3987
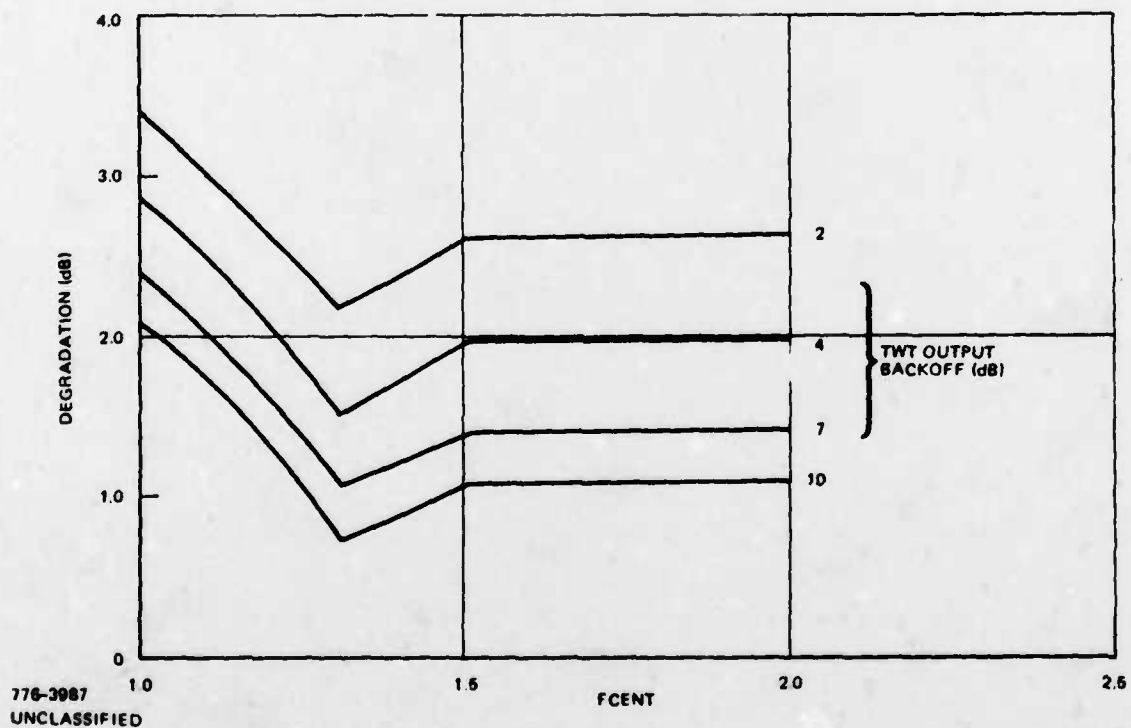
Figure 7-26. Degradation With Equalized Filters at E_b/N_o = 10 dB
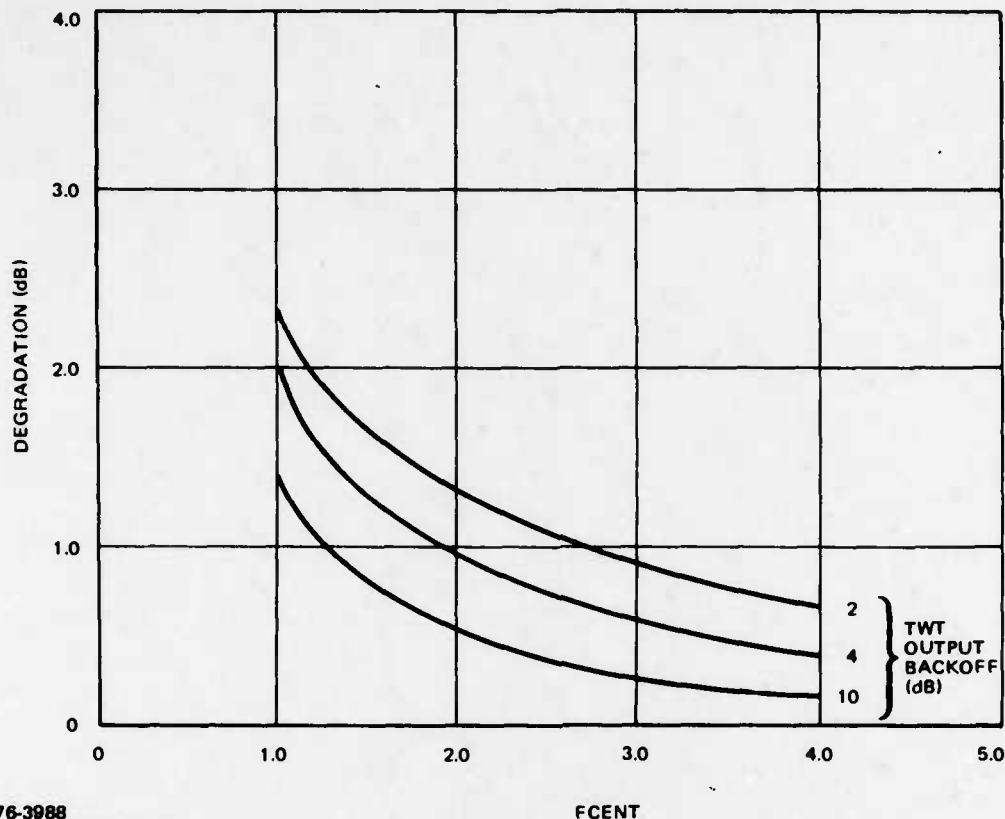
Figure 7-27. Degradation With No Filters at E_b/N_o = 2 dB

Comparing Figure 7-29 with Figure 7-26 shows that the predicted degradation in dB compares well with that E_b/N_o required to achieve 10^{-5} error probability for degradations less than 2 dB. For higher degradations, Figure 7-26 predicts less of an increase in E_b/N_o (at received E_b/N_o = 10 dB) than is actually required to achieve 10^{-5} probability of error. Figure 7-25 is not repeated for the lower error rate of .04 (at E_b/N_o = 2 dB) because it closely represents the actual increase in E_b/N_o required.

Figure 7-30 shows the effect of TWT output backoff on system performance for a fixed probability of bit error = 10^{-5}. The measure of system effectiveness is the sum of the output backoff (decrease in output power from saturation) and the increase in received E_b/N_o to achieve 10^{-5} error rate at that backoff. The curves are plotted for signal separations of 1.1, 1.3, and 1.5. As shown, a turnaround occurs in the range of output backoff between 2 dB and 3 dB, depending on signal separation. System performance deteriorates for backoffs less than this turnaround level. In effect, Figure 7-30 shows that, for backoffs below the turnaround level, the E_b/N_o required to achieve 10^{-5} error rate increases faster than the output power for a net loss in performance.
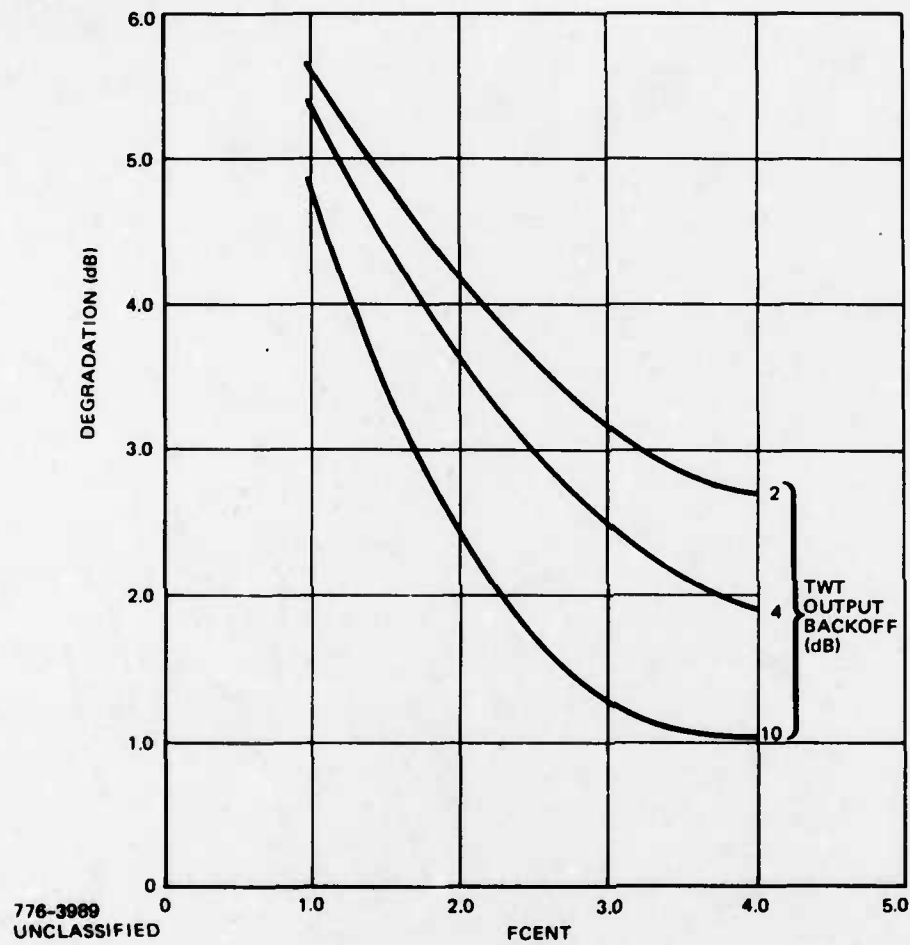
Figure 7-28. Degradation With No Filters at $E_b/N_o = 10$ dB
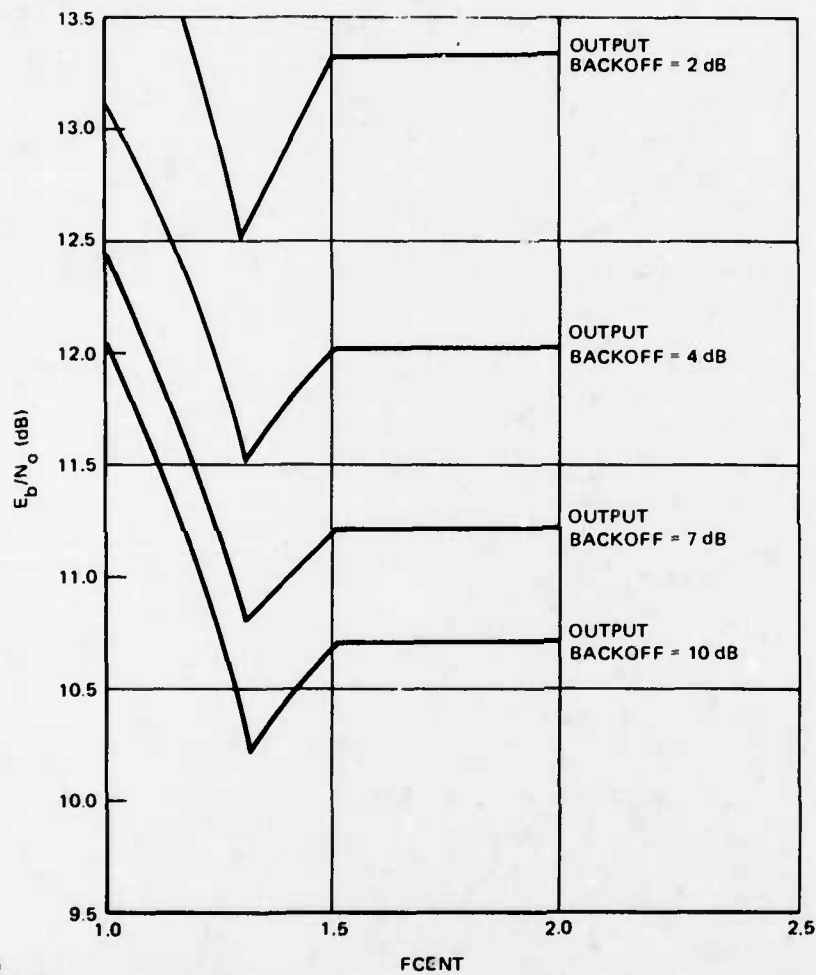
Figure 7-29. E_b/N_o Required to Achieve a Probability of Bit Error = 10^{-5} (Equalized Filters, Phase II TWT)
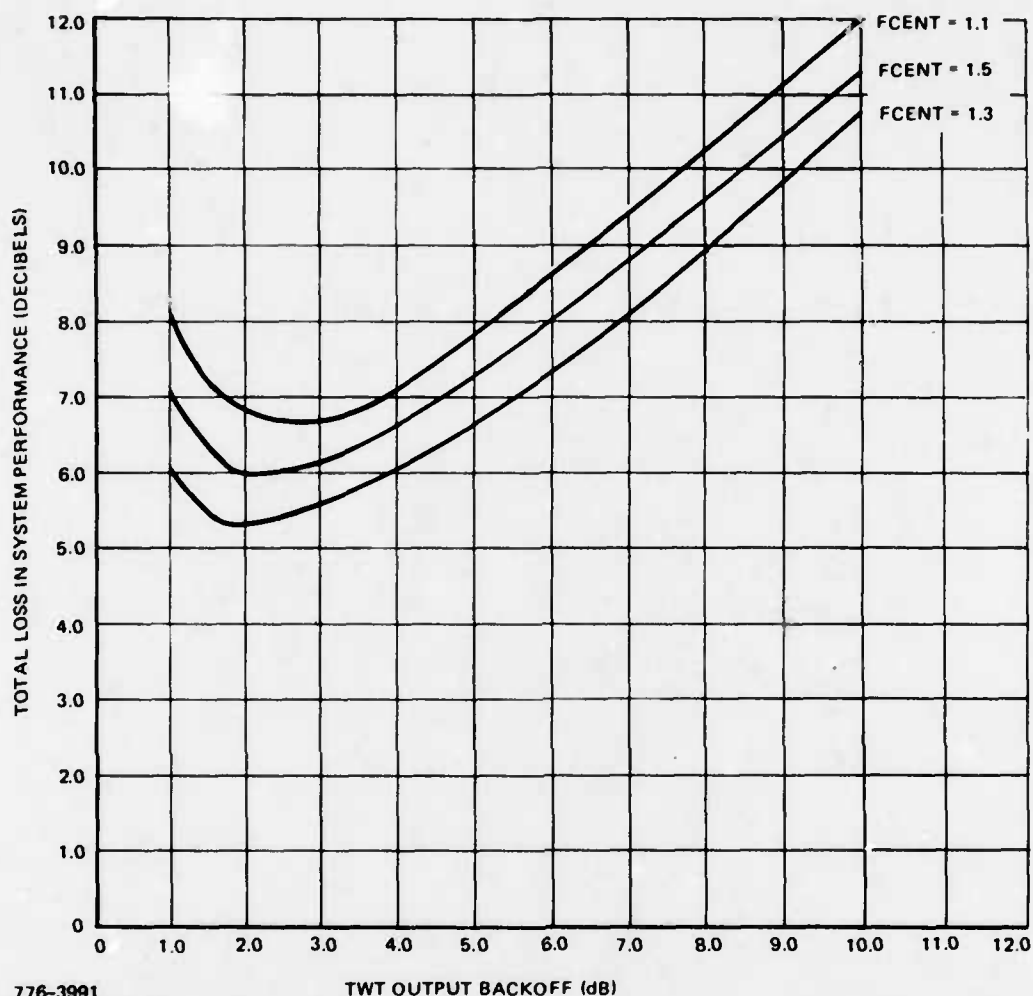
Figure 7-30. Total Loss in System Performance as a Function of TWT Output Backoff and Signal Separation (Equalized Filters, $P_e = 10^{-5}$)

If Figure 7-30 were plotted for lower probabilities of error, the turn-around would occur at higher output backoff levels. Similarly, for high probabilities of error, turnaround may not occur and the TWT would be operated at saturation for maximum system effectiveness.

Two considerations are of major interest in this investigation. One is the minimum signal separation possible for a given degradation. The other consideration involves how close to saturation the TWT can be operated. These considerations depend on the system filters, the E_b/N_o at which the system operates, and whether coding is used.

For E_b/N_o = 2 dB (coded operation), use of either Butterworth filters or equalized filters produces comparable degradations for FCENT > 1.5. For FCENT < 1.5, use of Butterworth filters produces degradation which monotonically increases as frequency separation decreases. On the other hand, for the equalized filter, degradation at a spacing FCENT = 1.2 is about the same as for FCENT > 1.5 but degradation rapidly increases for FCENT < 1.2. For E_b/N_o = 10 dB, and equalized filters degradation is significantly higher for all values of FCENT but operation at FCENT = 1.2 produces performance not appreciably worse than for large frequency separation. It appears that minimum degradation occurs if FCENT = 1.3.

Turnaround represents the backoff level at which the total system performance is best. As TWT output power is increased beyond this level, the required receiver E_b/N_o increases at a more rapid rate than the output power for a net loss in performance. For high probabilities of bit error, such as that in the range of E_b/N_o where coding would be employed, no turnaround exists and the TWT should be operated as close to saturation as feasible. For low probabilities of bit error, a turnaround point exists where system effectiveness is a maximum and the TWT drive should be adjusted to this level. The exact turnaround drive is somewhat dependent on signal spacing. For a probability of bit error = 10^{-5}, the optimum output backoff is between 2 and 3 dB. This optimum backoff increases as the desired probability of error is reduced. Coding can be used to obtain a lower error rate at a given backoff value but at a cost of increasing the signal separation due to the bandwidth expansion.

7.3.3   PERFORMANCE WITH HUGHES' TWT: EQUAL SIGNAL LEVELS

The Hughes' TWT will be used in the Phase III satellite. It is of interest to compare the performance of this amplifier with that used in the Phase II satellite. Five equal level signals simultaneously access the satellite. The power differential of 0 dB represents the desired condition of operation for the ground mobile forces application.

Only the equalized filters (Figure 7-3) are used in the transmitter and receiver based on their superiority over Butterworth filters for close signal spacing. All backoffs refer to average output power backoff. The peak power for the cases run is 7.0 dB above this level so that with the TWT model used, the signal peaks are clipped in amplitude for backoffs less than 7.0 dB. Similarly, the signal peaks receive a phase shift of 66° for backoffs less than 7.0 dB.

Figures 7-31 through 7-33 show the degradation produced by the Phase III TWT as a function of output power backoff and signal spacing. Figures 7-31 and 7-32 show a decrease in degradation for FCENT around 1.3. This effect appears due to the equalized filter peak and its interaction with the adjacent signal intermodulation spectrum. At a spacing FCENT ' 1.5, this adjacent peak is accentuated by the receiver filter peak. At a spacing FCENT ≐ 1.3, the receive filter peak falls over the adjacent signal spectral null.

Figure 7-33 shows the effect of TWT output backoff on system performance for a fixed probability of bit error $= 10^{-5}$. In the ideal channel, an $E_b/N_o \cong 9.6$ dB achieves this error rate. The measure of loss in system performance is the sum of the output backoff (decrease in output power from saturation) and the increase in received E_b/N_o to achieve 10^{-5} error rate at that backoff. The curves are plotted for signal separations of 1.1, 1.3, and 1.5. As shown, a turnaround occurs at about an output backoff of 3 dB, depending on signal separation. System performance deteriorates for backoffs less than this turnaround level. In effect, Figure 7-33 shows that for backoffs below the turnaround level, the E_b/N_o required to achieve 10^{-5} error rate increases faster than the output power for a net loss in performance.
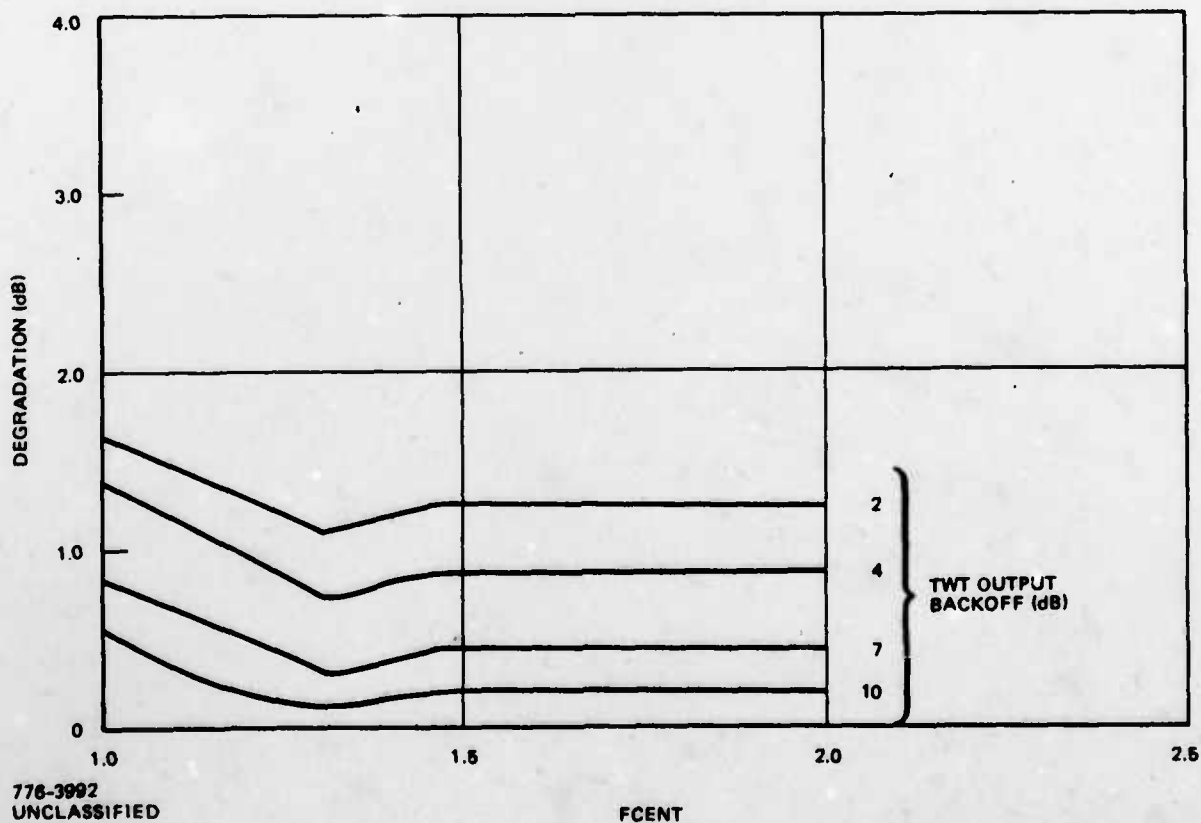
Figure 7-31. Degradation With Phase III Satellite TWT at $E_b/N_o = 2$ dB

7-39
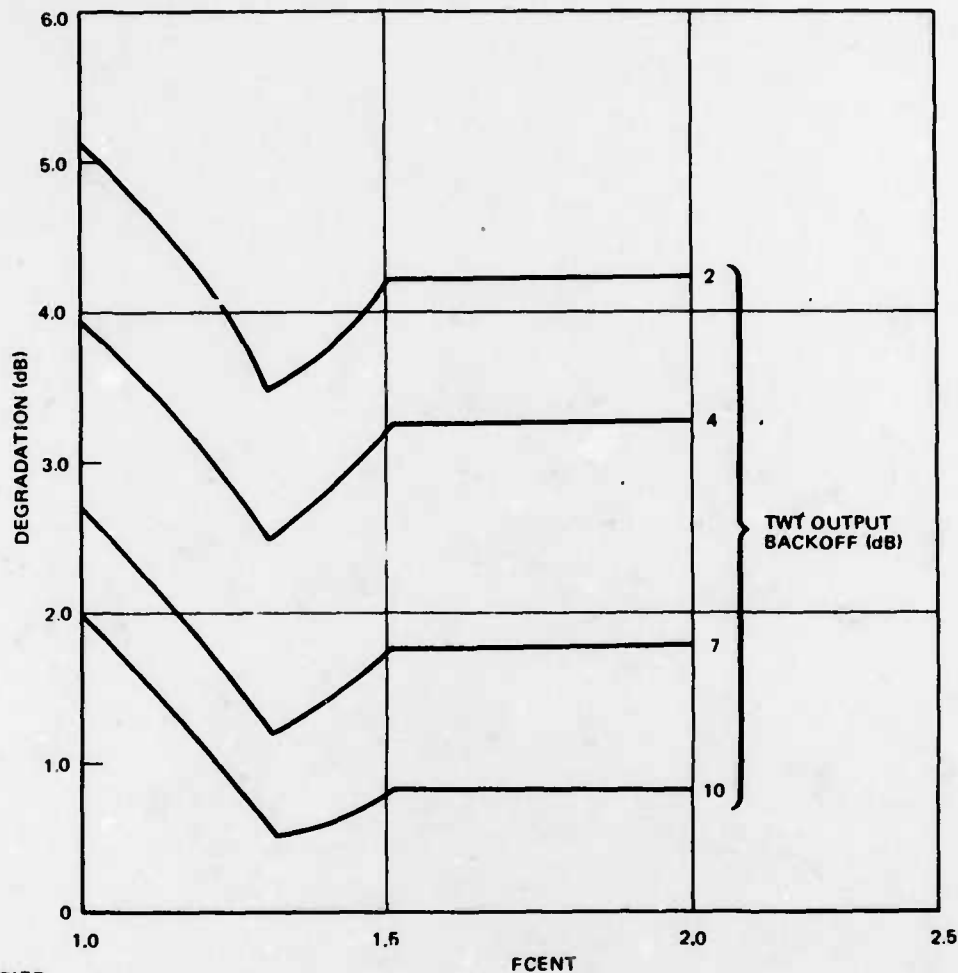
Figure 7-32. Degradation With Phase III Satellite at $E_b/N_o = 10$ dB

If Figure 7-33 were plotted for lower probabilities of error, the turn-around would occur at higher output backoff levels. Similarly, for high probabilities of error, turnaround may not occur and the TWT would be operated at saturation for maximum system effectiveness.

For $E_b/N_o = 2$ dB (coded operation), degradation at a spacing FCENT = 1.2 is about the same as for FCENT > 1.5 but degradation rapidly increases for FCENT < 1.2. For $E_b/N_o = 10$ dB, degradation is significantly higher for all values of FCENT but degradation at FCENT = 1.2 is not appreciably higher than for large frequency separation. Minimum degradation occurs if FCENT = 1.3.
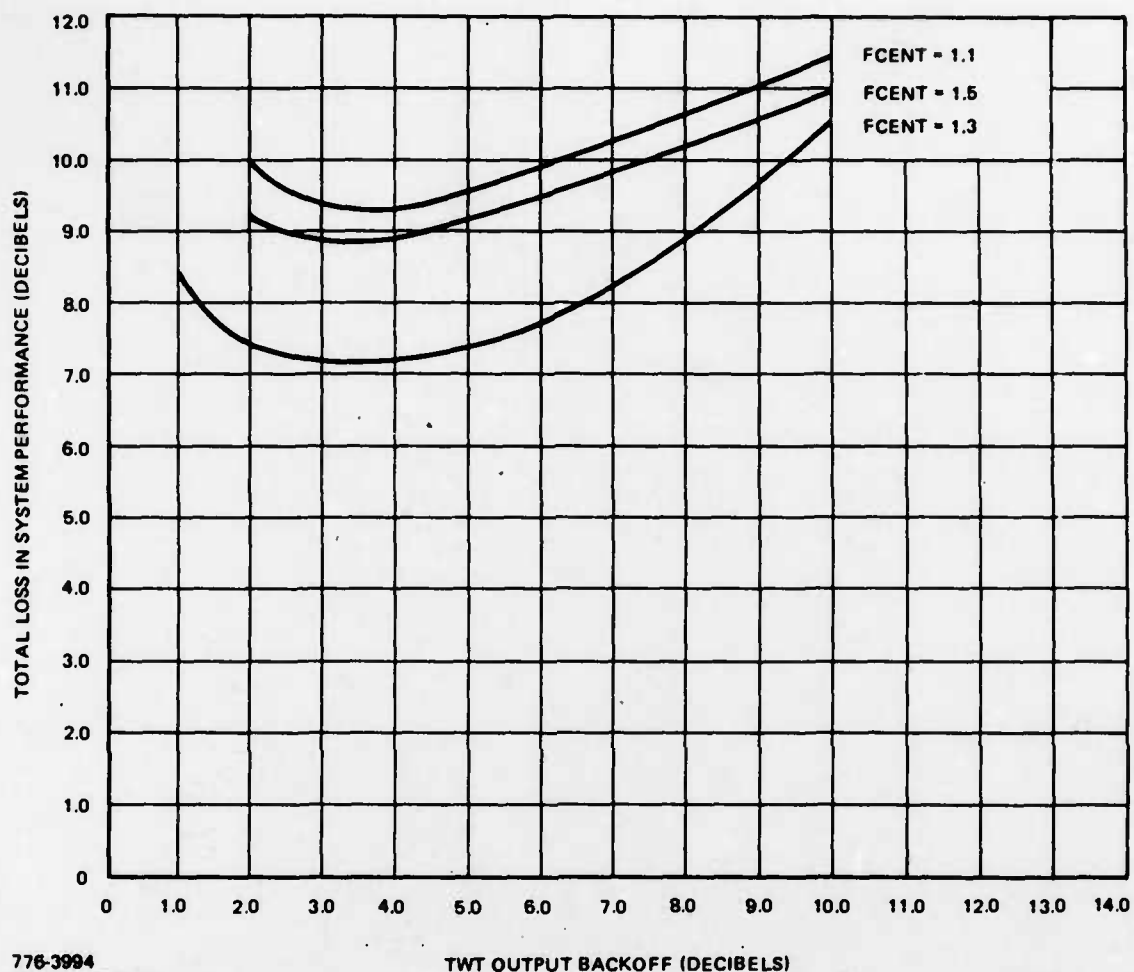
Figure 7-33. Total Loss in System Performance as a Function of Phase III TWT
Output Backoff and Signal Separation (Equalized Filters, $P_e = 10^{-5}$)

For low probabilities of bit error, a turnaround point exists where system
effectiveness is a maximum and the TWT drive should be adjusted to this level. The
exact turnaround drive is somewhat dependent on signal spacing. For a probability
of bit error = 10^{-5}, the optimum output backoff is about 3 dB. This optimum backoff
increases as the desired probability of error is reduced. Coding can be used to
obtain a lower error rate at a given signal separation and backoff value. Section 7.4
gives results for the coded saturating channel.

Comparing the results in this study with the corresponding results in
7.3.2 using the Phase II TWT shows an increase in degradation for the Hughes' TWT
at the lower values of TWT backoff. Presumably, this is predominantly due to the

increased AM-PM conversion of the Hughes' TWT as compared to the Phase II TWT. For large output backoff, the Hughes' TWT produces somewhat less degradation than the Phase II TWT. The total loss in system performance, at $P_e = 10^{-5}$, was minimized at an output backoff of 2 to 3 dB for the Phase II TWT and at an output backoff of 3 to 4 dB for the Hughes' TWT.

7.3.4 PERFORMANCE WITH PHASE II TWT: UNEQUAL SIGNAL LEVELS

The results of Section 7.3.2 are repeated for the case of all four adjacent signals having power 6 dB greater than that of the center signal. The power differential of 6 dB represents a worst case condition of operation for the ground mobile forces application. Note that since the center channel power is now 6 dB less than each of the adjacent channels, the peak signal-to-intermodulation ratio in the center channel, shown in Figure 7-21, now becomes 6 dB at a TWT output backoff of 4 dB.

Figures 7-34 through 7-37 show the degradation produced by the Phase II TWT as a function of output power backoff and signal spacing. Figures 7-34 and 7-35 assume use of the 5-pole zero phase Butterworth filters at $E_b/N_o = 2$ dB and 10 dB. Figures 7-36 and 7-37 assume use of the equalized filters at $E_b/N_o = 2$ dB and 10 dB.

776-3995
UNCLASSIFIED
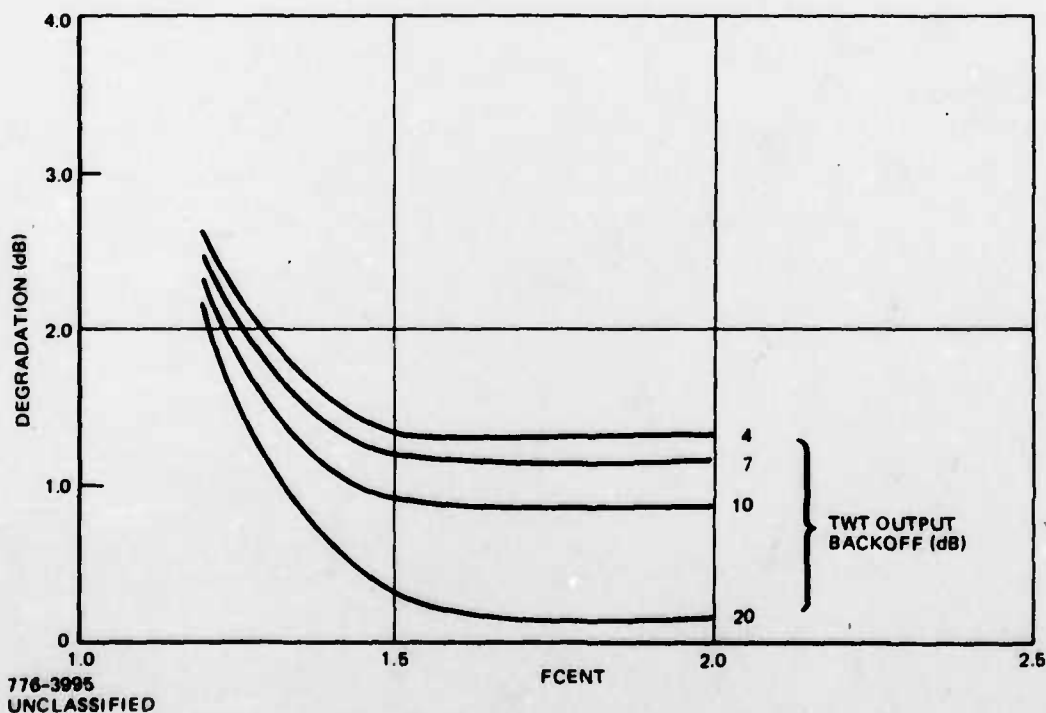
Figure 7-34. Degradation With TWT and Butterworth (BT = .75) Filters at $E_b/N_o = 2$ dB
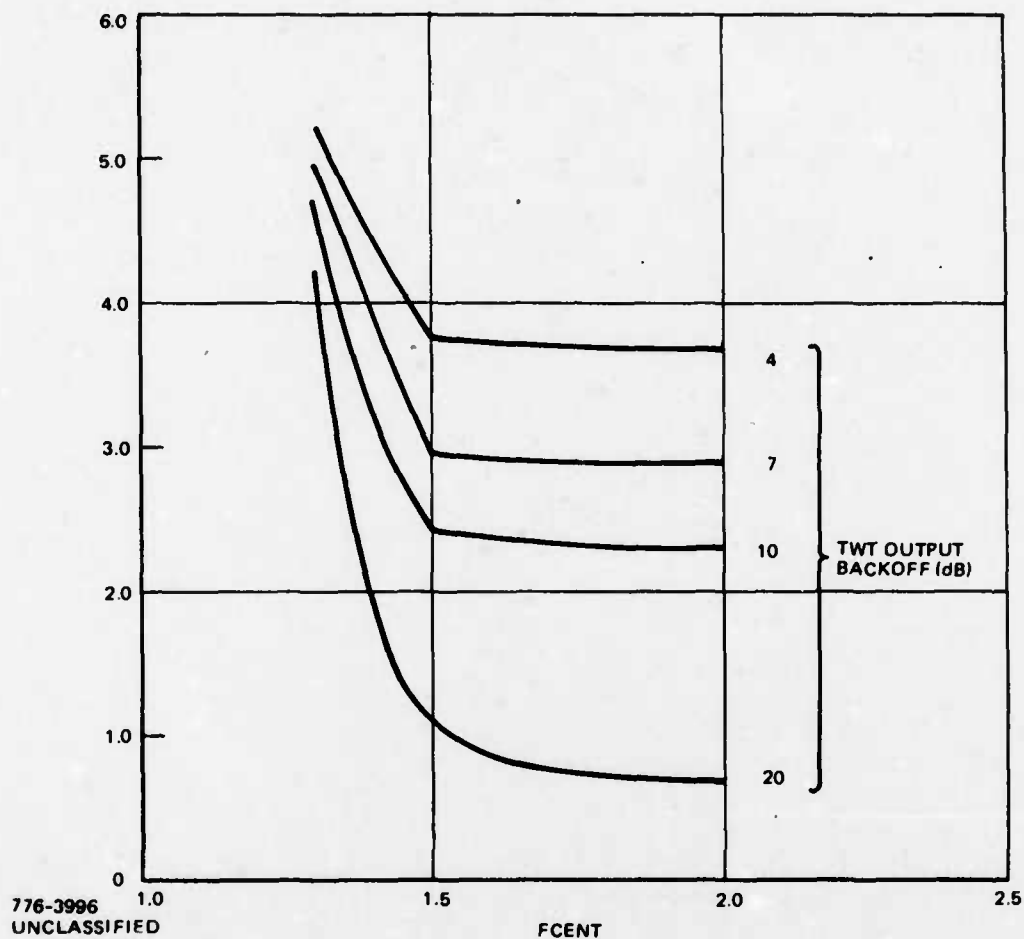
Figure 7-35. Degradation With TWT and Butterworth (BT_b = .75) Filters at E_b/N_o = 10 dB

Figures 7-36 and 7-37 show a decrease in degradation for FCENT around 1.3. This dip does not occur in Figures 7-34 and 7-35.

776-3997
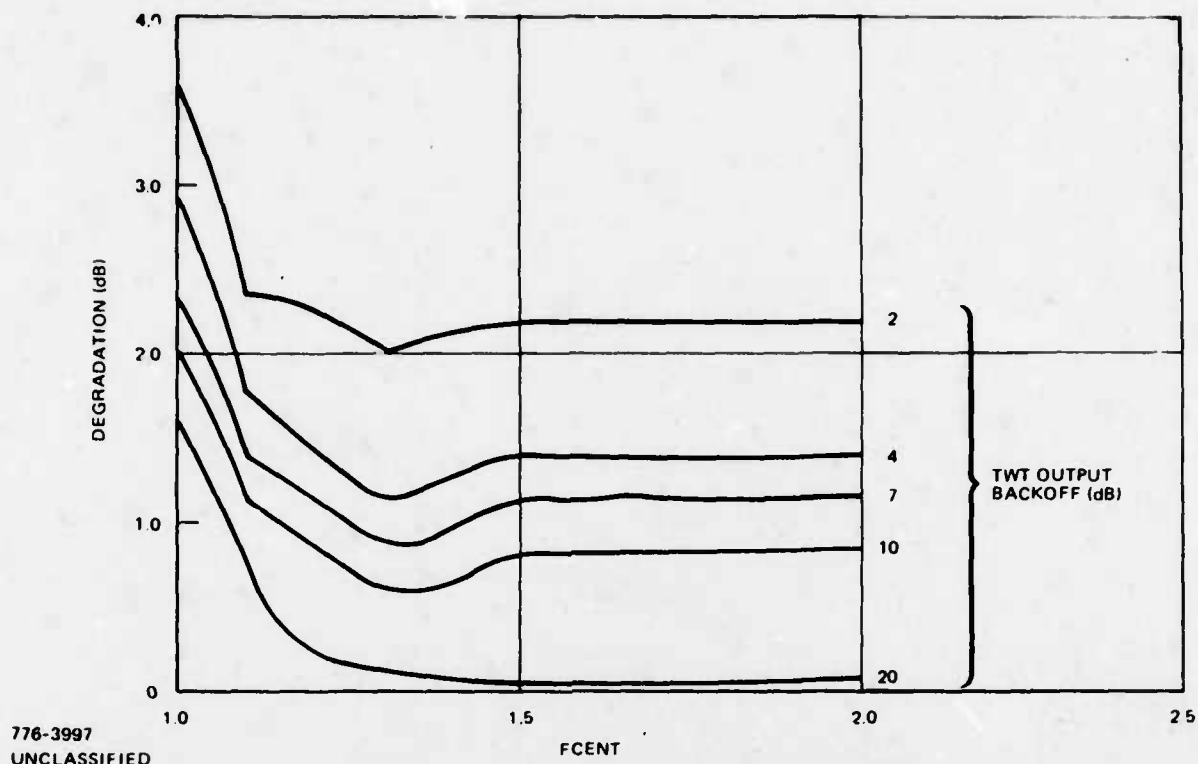
Figure 7-36. Degradation With TWT and Equalized Filters at $E_b/N_o = 2$ dB

For $E_b/N_o = 2$ dB (coded operation), use of either Butterworth filters or equalized filters produces comparable degradations for FCENT > 1.5. For FCENT < 1.5, use of Butterworth filters produces degradation which monotonically increases as frequency separation decreases. On the other hand, for the equalized filter, degradation at a spacing FCENT = 1.1 is about the same as for FCENT > 1.5 but degradation rapidly increases for FCENT < 1.1. For $E_b/N_o = 10$ dB, and equalized filters degradation is significantly higher for all values of FCENT but operation at FCENT = 1.2 produces degradation not appreciably higher than for large frequency separation. It appears that minimum degradation occurs if FCENT = 1.3.

For operation near saturation, decreasing the backoff results in less gain in received E_b/N_o than a corresponding change at higher backoff. For example, Figure 7-36 shows that decreasing backoff from 7 dB to 4 dB gives a net effective gain in received E_b/N_o of about 2.8 dB. However, decreasing backoff from 4 dB to 2 dB gives a net effective gain in received E_b/N_o of 1.2 dB due to the increased degradation.

DEGRADATION (dB)

6.0

5.0

4.0

3.0

2.0

1.0

0

4

7

10

20

TWT OUTPUT
BACKOFF (dB)

1.0   1.5   2.0   2.5

FCENT
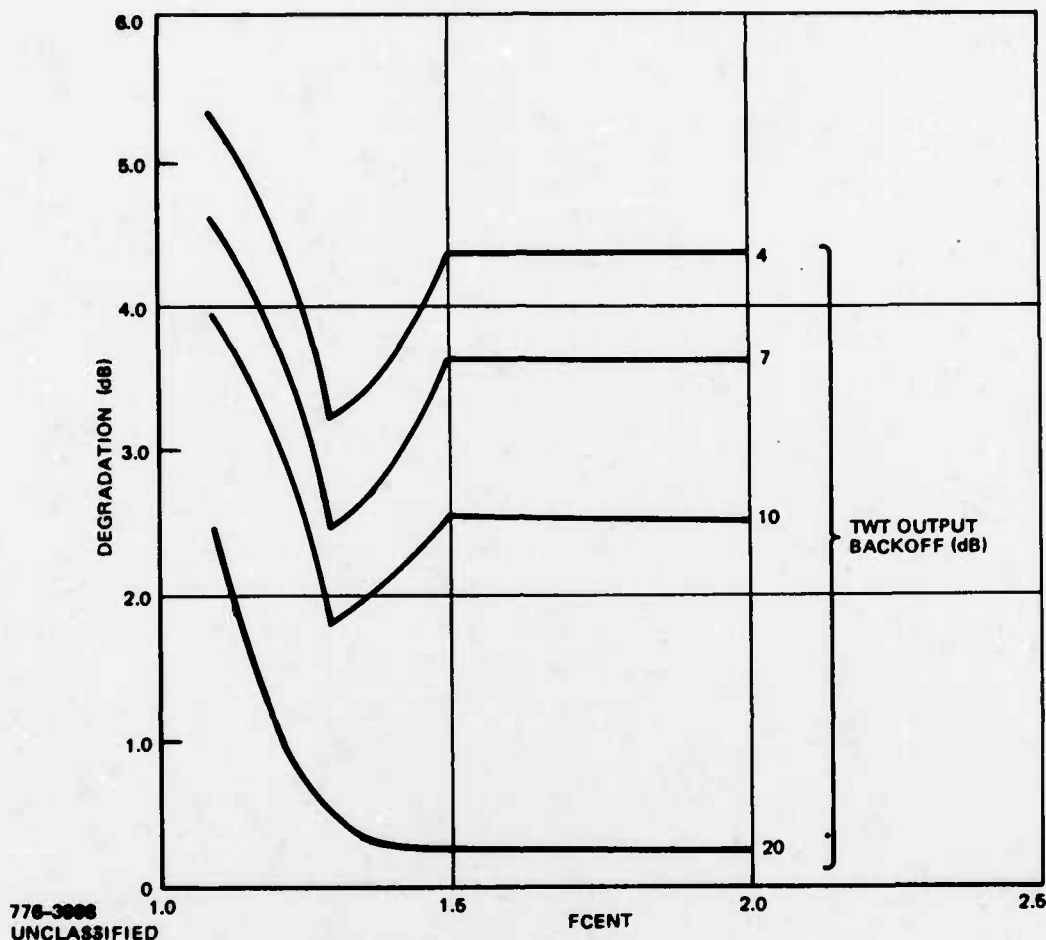
Figure 7-37.   Degradation With TWT and Equalized Filters at E_b/N_o = 10 dB

The increased intermodulation also places a floor on achievable probability of error no matter how low the receiver noise. If the signal-to-intermodulation ratio were 6 dB, for example, and the signal-to-thermal noise ratio at the receiver were 10 dB, the error rate would be predominantly limited by the intermod level. Therefore, backoff may be limited by the error rate requirements. To achieve low error rates while operating the TWT close to saturation, coding is required. Coding thus permits operation near saturation, close spacing of signals, and an acceptable error rate.
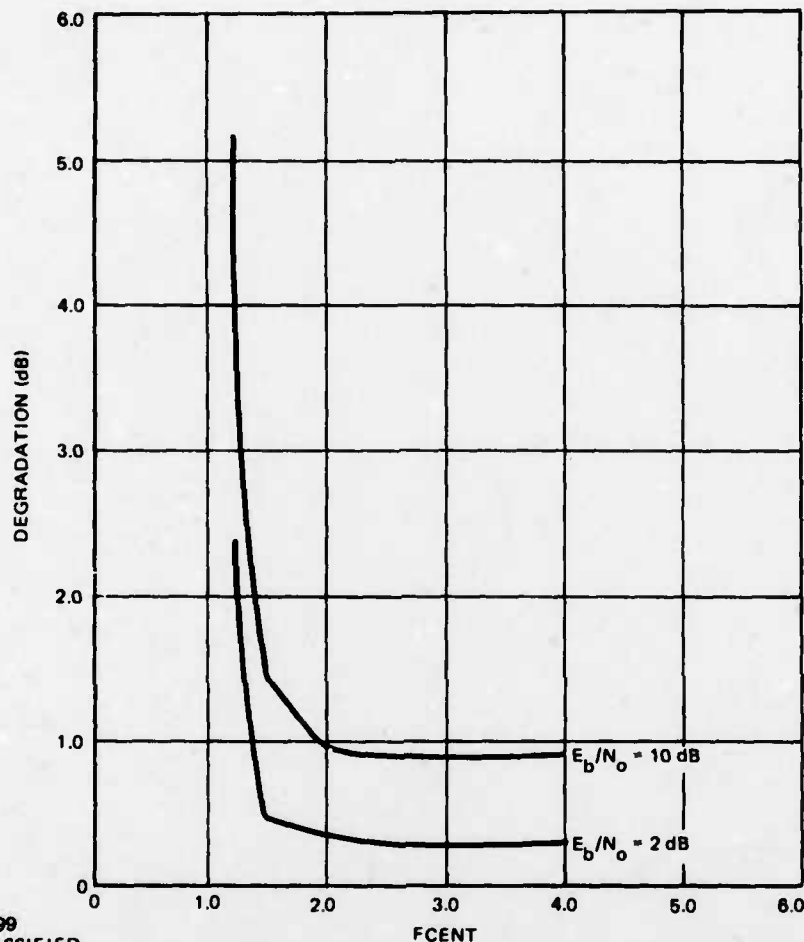
Figure 7-38. System Degradation as a Function of Signal Spacing With Hard Limiter Following Transmit Filter (Butterworth, $BT_b = .75$)

Figures 7-38 and 7-39 show the degradation occuring with each of the individual signals hard limited following the transmit filter. No satellite TWT is used. Figure 7-38 is based on use of the 5-pole Butterworth filters ($BT_b = .75$) while Figure 7-39 assumes equalized filters. Both curves show a rapid increase in degradation for FCENT < 1.5 with the Butterworth filters producing slightly less degradation for all signal separations. The hard limiters have changed the signal spectral characteristics to such a degree that the filter equalization characteristic is no longer effective as it is in a quasilinear channel.

DEGRADATION (dB)

$E_b/N_o = 10$ dB

$E_b/N_o = 2$ dB

FCENT
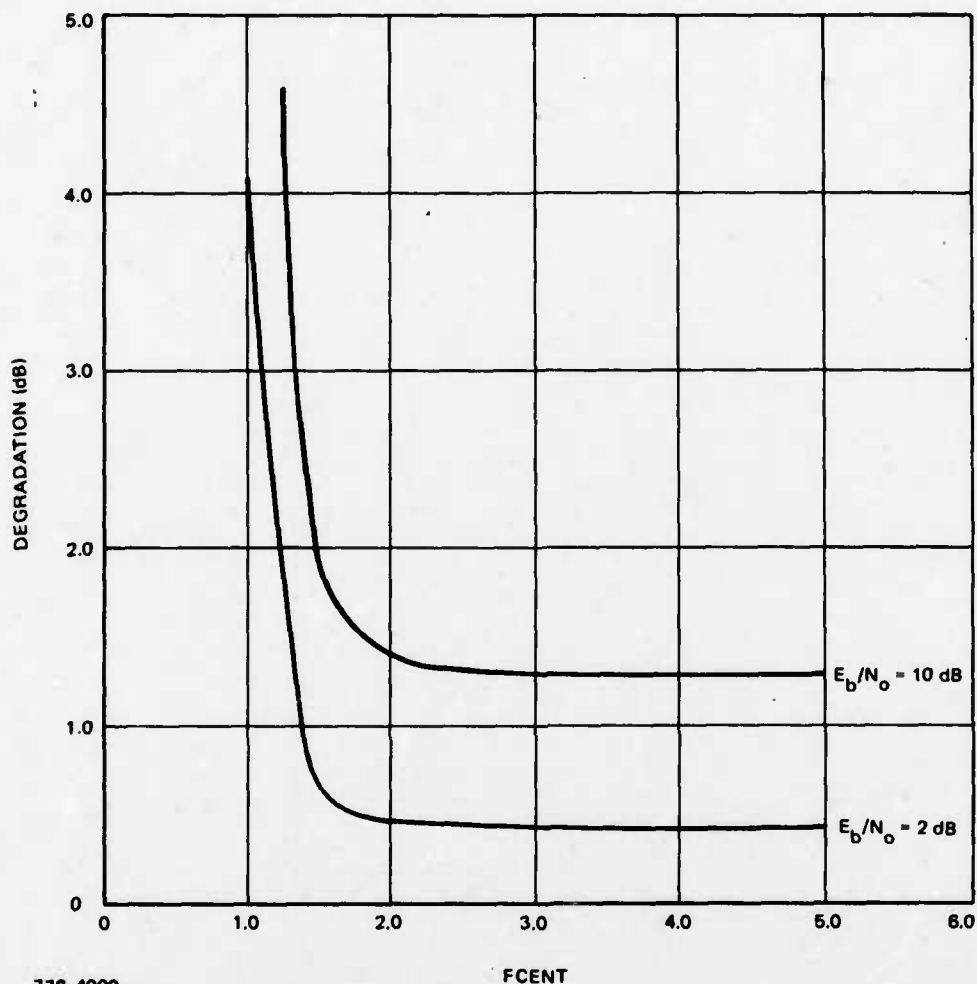
Figure 7-39. System Degradation as a Function of Signal Spacing With Hard
Limiter Following Transmit Filter (Equalized Filter)

7.3.5   PERFORMANCE WITH HARD LIMITING TRANSMITTERS AND PHASE
II TWT IN SATELLITE

Sections 7.3.2 through 7.3.4 evaluated the degradation occuring with
multiple accessing signals (FDMA) in a saturating channel where either a hard
limiting transmitter or a satellite TWT is present. This section evaluates the per-
formance of the FDMA system when both hard limiting transmitters and a Phase II
satellite TWT are operating in the channel.

The case of five equally-spaced and equal power signals is evaluated.
Equalized filters are used in the transmitter (before hard limiting) and in the receiver.
Hard limiters follow each of the transmit filters shown in Figure 7-18.

Figure 7-40 shows degradation for a received E_b/N_o = 2 dB and Figure
7-41 shows degradation for a received E_b/N_o = 10 dB. Both curves show a rapid
increase in degradation for FCENT < 1.5. No dip in degradation occurs at FCENT =
1.3 as is produced when equalized filters are used in conjunction with a TWT ampli-
fier in the satellite. This is due to the change in the signal spectral characteristics
produced by the hard limiter. As a result, the equalized filter characteristic is no
longer effective as it is in a quasilinear channel.

Inclusion of a hard limiter in the transmitters following the filter increases
the minimum spacing of the FDMA signals. This is due to the spectral broadening of
the individual signals produced by the hard limiters. No benefits are produced by the
equalized filters in the hard limited channel as occured in the quasilinear channel, but
no more degradation is produced than would occur with Butterworth filters. Coding
is of significant benefit if low error rates are desired in a channel with hard limiting,
as shown in Section 7.4.

776-4001
UNCLASSIFIED
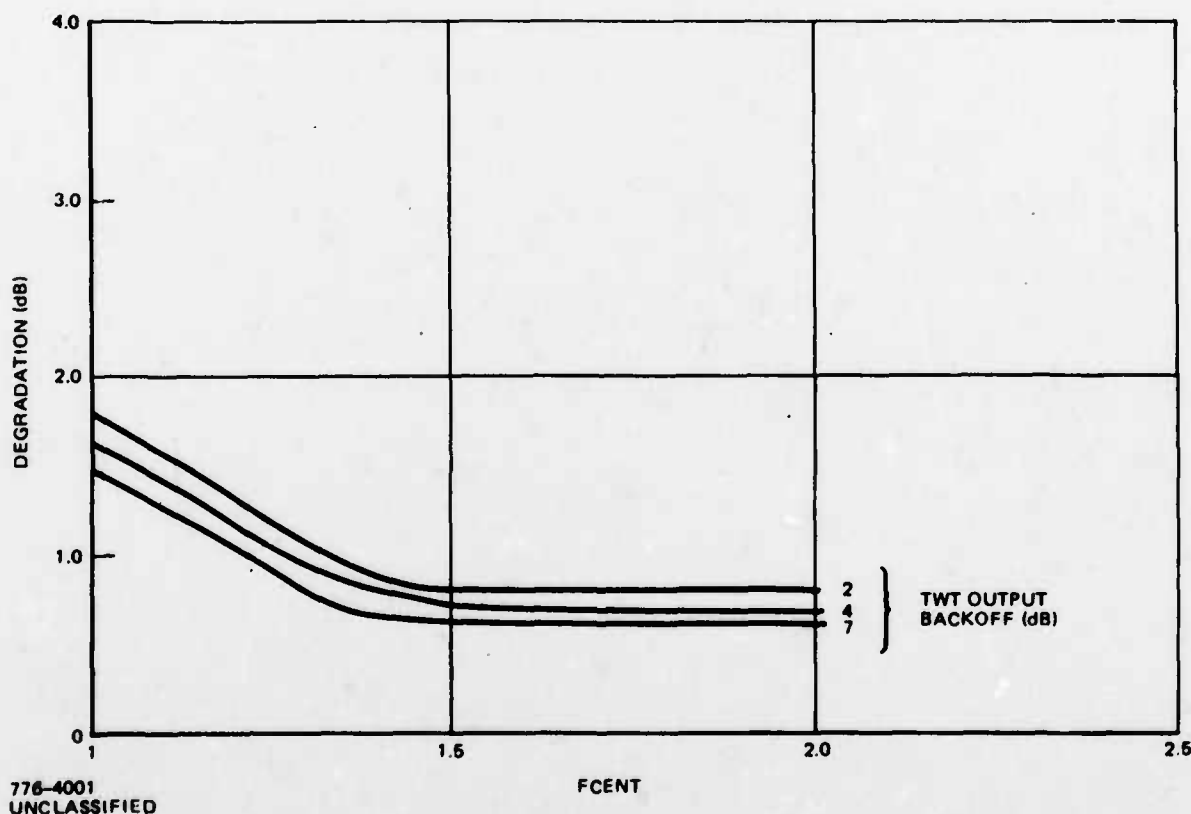
Figure 7-40. Degradation Produced by Hard Limiting Transmitters and Phase
II TWT at E_b/N_o = 2 dB

7-48
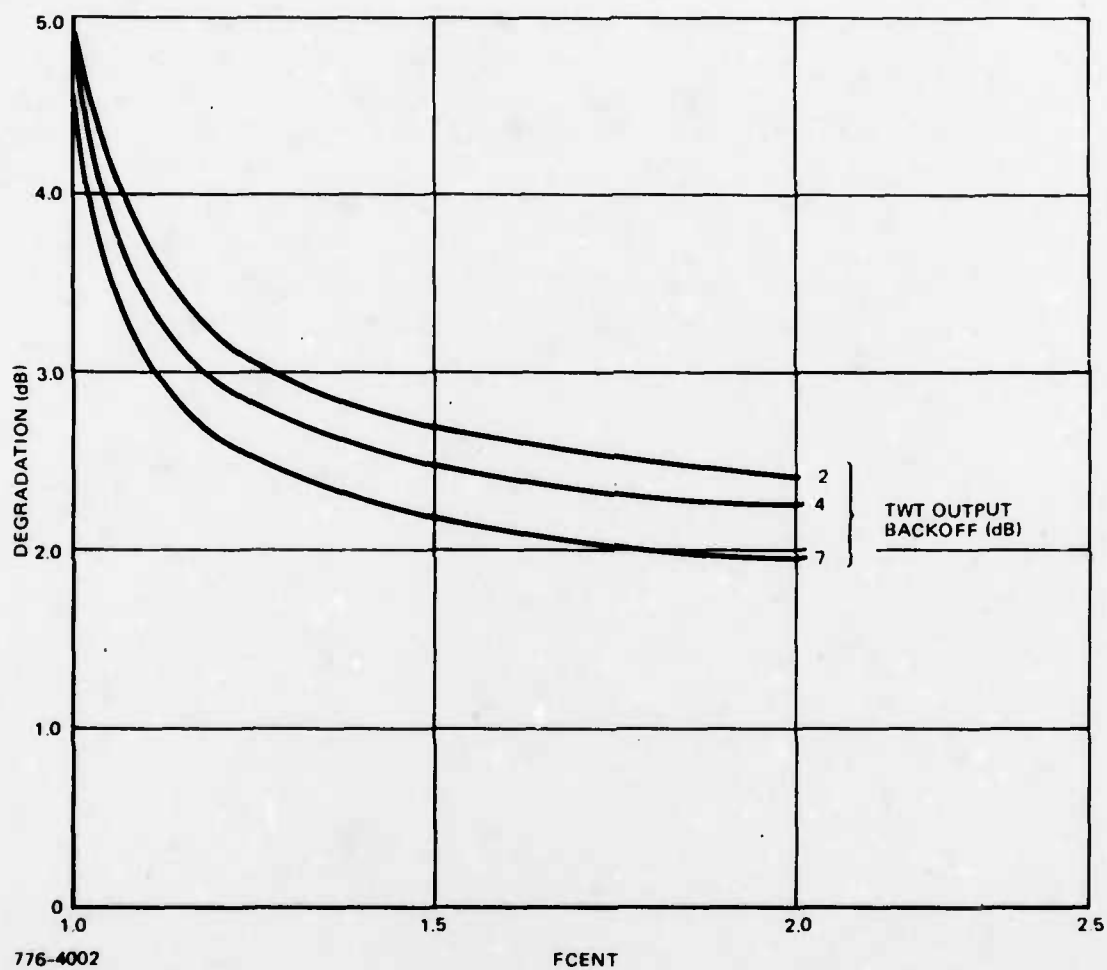
Figure 7-41. Degradation Produced by Hard Limited Transmitters and Phase II
TWT at $E_b/N_o = 10$ dB

7.3.6 UNCODED FDMA: PERFORMANCE COMPARISONS

Sections 7.3.1 through 7.3.5 contain results based on passing five (equal or unequal levels) SQPSK signals through the Phase II or anticipated Phase III TWT repeater channel. The effect of varying the frequency spacing was emphasized. This section presents these results in a series of curves giving the performance degradation* as a function of received E_b/N_o at a fixed frequency spacing for each case considered. The spacing is selected as that where adjacent channel interference is just becoming noticeable. Extension of the SQPSK results to QPSK and BPSK is also discussed.

As in the preceding sections, the five accessing signals are equally spaced and degradation is evaluated for the center signal. Identical linear-phase Butterworth or equalized (amplitude-shaped) filters are placed in the transmitter and the receiver. The equalized filters have $BT_b = .5$ and the 5-pole Butterworth filters have $BT_b = .75$. As mentioned in Section 7.3.2, high degradation occurs when no filters are used for the close spacings and TWT backoffs of interest; therefore, curves for the case of no transmitter or receiver filters are not included.

The chosen frequency spacing (FCENT) for use with the Butterworth filters is 1.5 while the spacing used with the equalized filters is 1.25. (Note, FCENT 1.0 specifies a frequency spacing equal to the bit rate for uncoded BPSK, or half the total bit rate for uncoded QPSK or SQPSK.) These frequency spacings produce degradation levels at all values of backoff and E_b/N_o of interest that are no worse than the system degradation produced for large spacings. Performance deteriorates rapidly for closer signal spacings. For low values of backoff (or received E_b/N_o), somewhat closer signal spacings could be used.

Figures 7-42 through 7-45 show the system degradations for the case of equal-level accessing signals. Figures 7-42 and 7-43 are based on the Phase II TWT using Butterworth or equalized filters. Figures 7-44 and 7-45 assume the Phase III (Hughes') TWT characteristic, and Butterworth or equalized filters.

Figures 7-46 through 7-49 give the corresponding system degradations for the case of unequal-level accessing signals. The center signal is 6 dB below each of the four adjacent channels. This power differential for signal accesses represents an assumed worst case condition of operation for the ground mobile forces application.

---

*Degradation is defined as the reduction in E_b/N_o to obtain the same error rate with ideal BPSK or QPSK. Note that degradation tends to increase as the error rate decreases.

All curves are given for the case of uncoded SQPSK signals as a function of E_b/N_o. The use of QPSK yields performance virtually identical with that resulting from SQPSK. QPSK degradation is slightly less, but always within 0.25 dB of that obtained with SQPSK. The same frequency spacing should be used with QPSK as for SQPSK. Biphase (BPSK) operation also produces no major differences in degradation over SQPSK and the same recommended signal spacings should be used.

As will be seen in Section 7.4, the degradation for rate-1/2 coded SQPSK can be estimated from Figures 7-42 to 7-49 for uncoded data by reading the curves at $E_b/N_o = 2$ dB. Because of the code redundancy, this operating point is equivalent to $E_b/N_o = 5$ dB, defined with respect to a data bit. Note that FCENT = 1.0 then specifies a frequency spacing equal to the actual data rate, with rate-1/2 coded SQPSK.
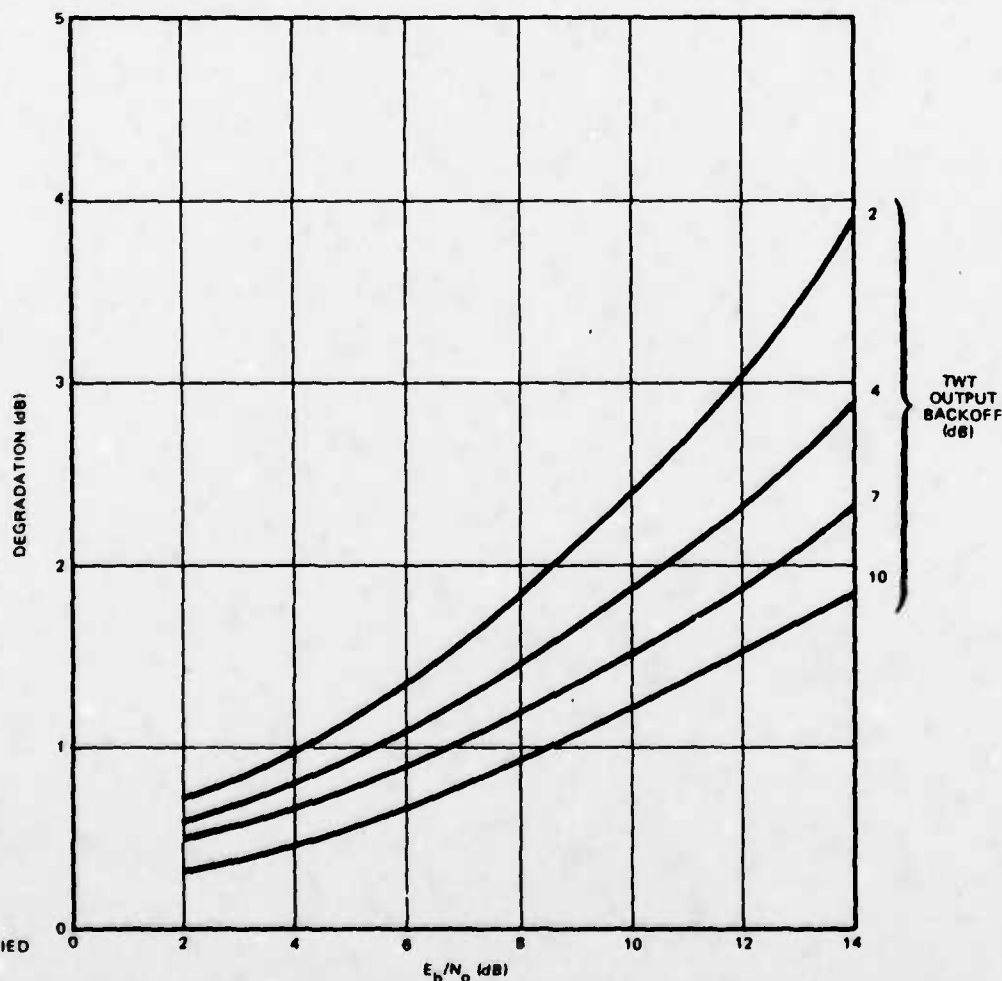
Figure 7-42. Degradations versus Received E_b/N_o (Equal Signal Levels): Phase II TWT, Butterworth Filters ($BT_b = .75$)
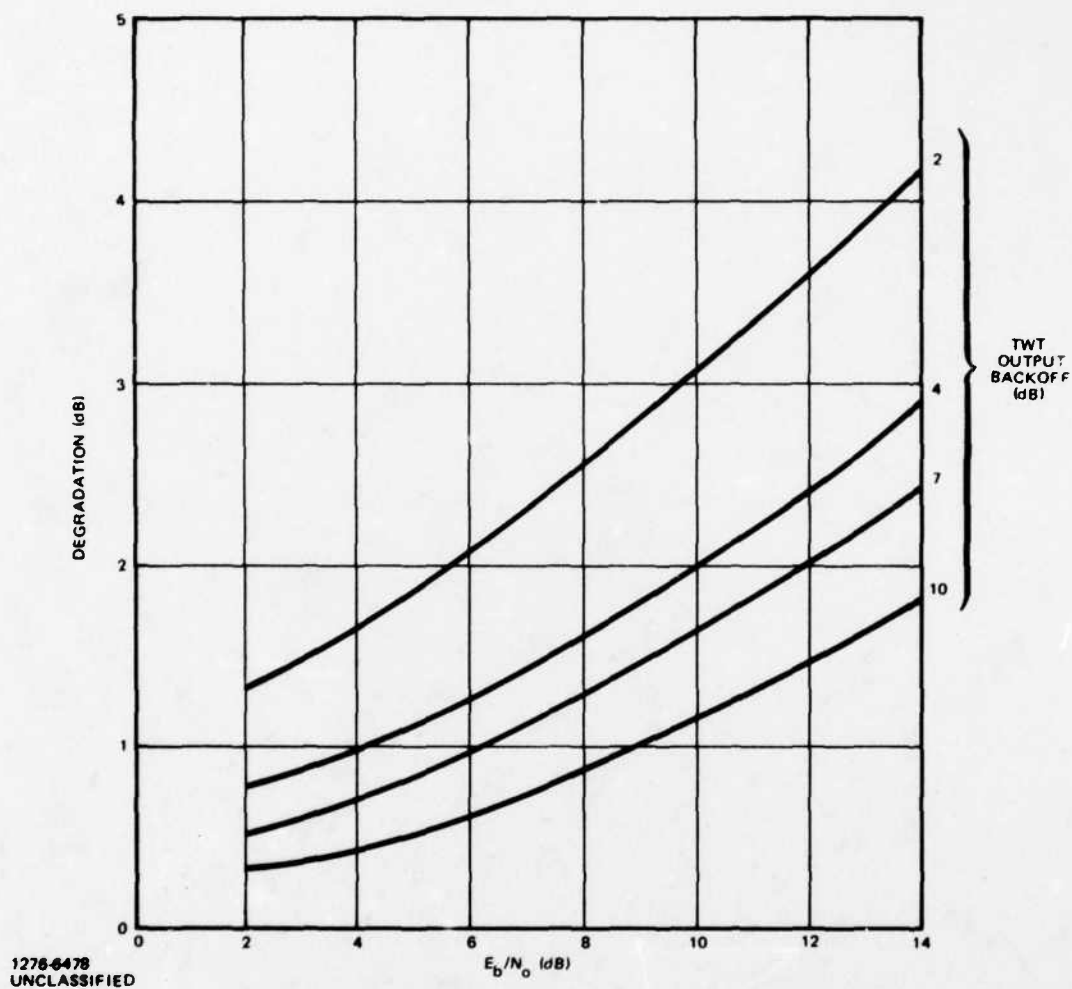
Figure 7-43. Degradation versus Received E_b/N_o (Equal Signal Levels): Phase II
TWT, Equalized Filters
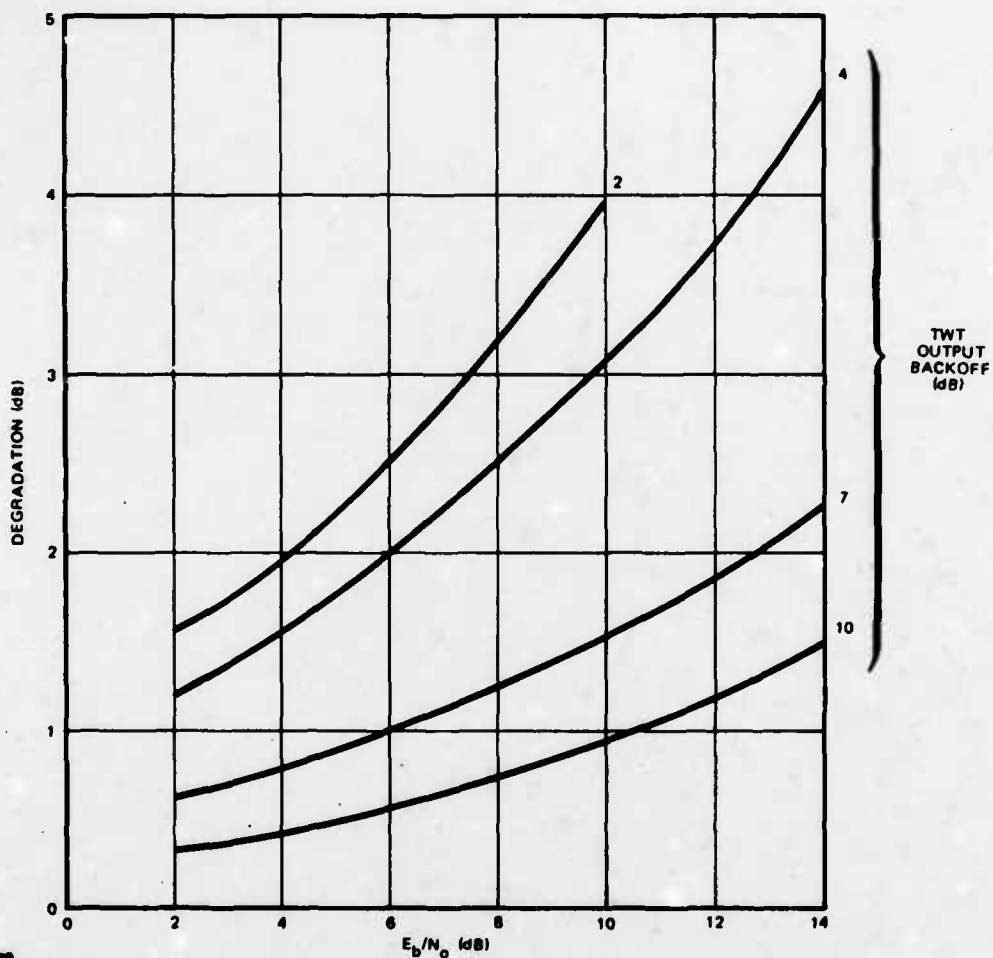
Figure 7-44. Degradation versus Received E_b/N_o (Equal Signal Levels): Phase III TWT, Butterworth Filters ($BT_b = .75$)
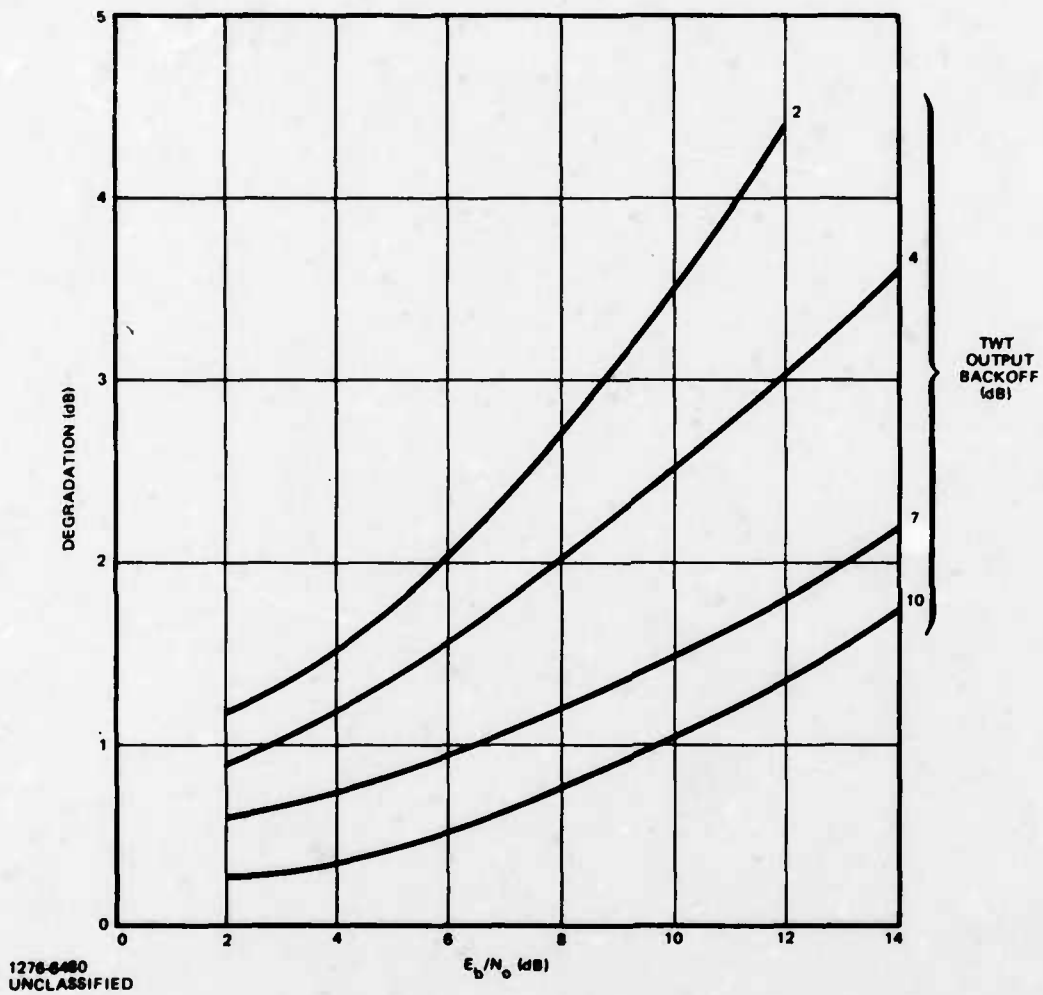
Figure 7-45. Degradation versus Received E_b/N_o (Equal Signal Levels): Phase III TWT, Equalized Filters
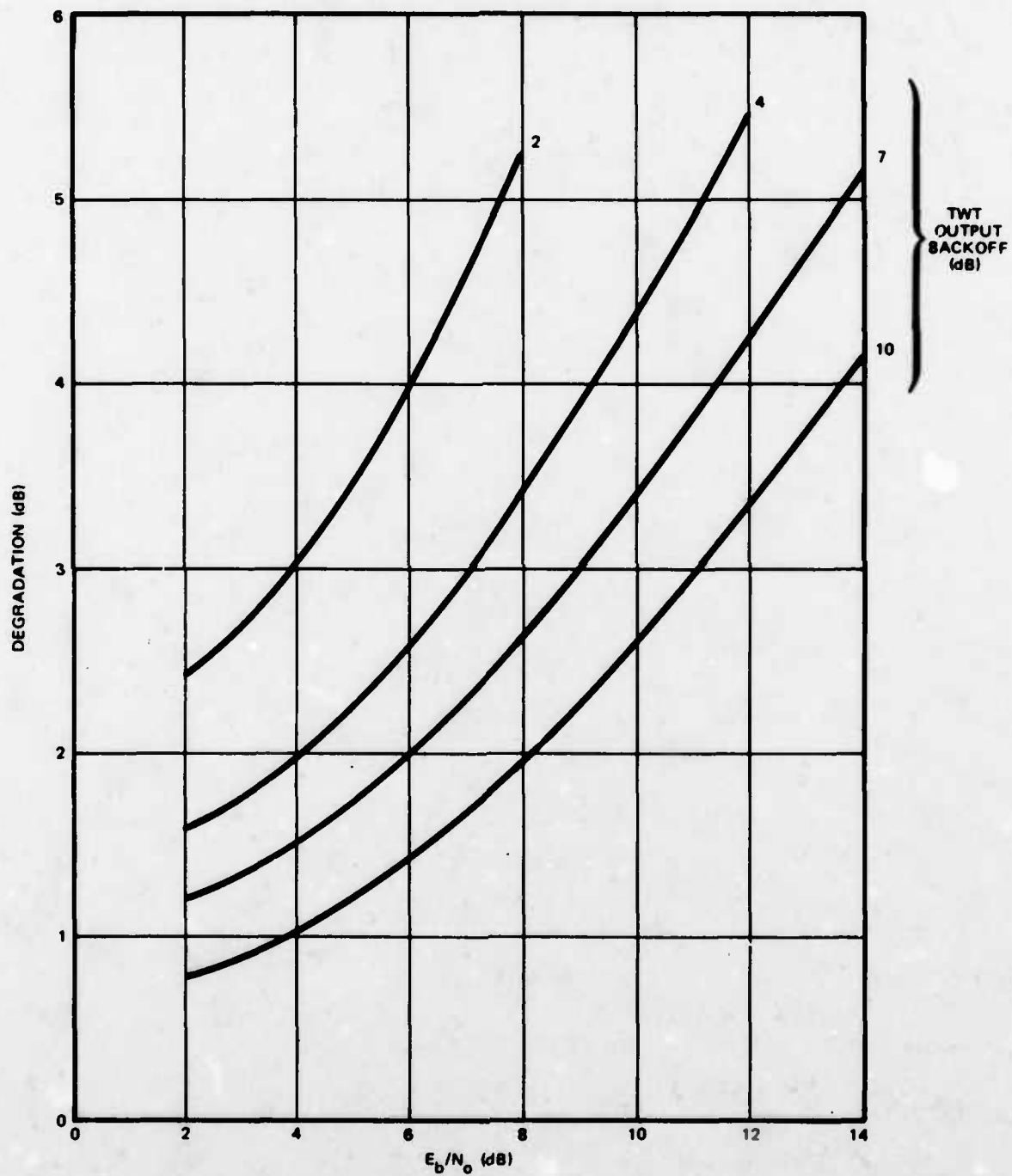
Figure 7-46. Degradation versus Received E_b/N_o (Unequal Signal Levels): Phase II
TWT, Butterworth Filters (BT_b = .75)
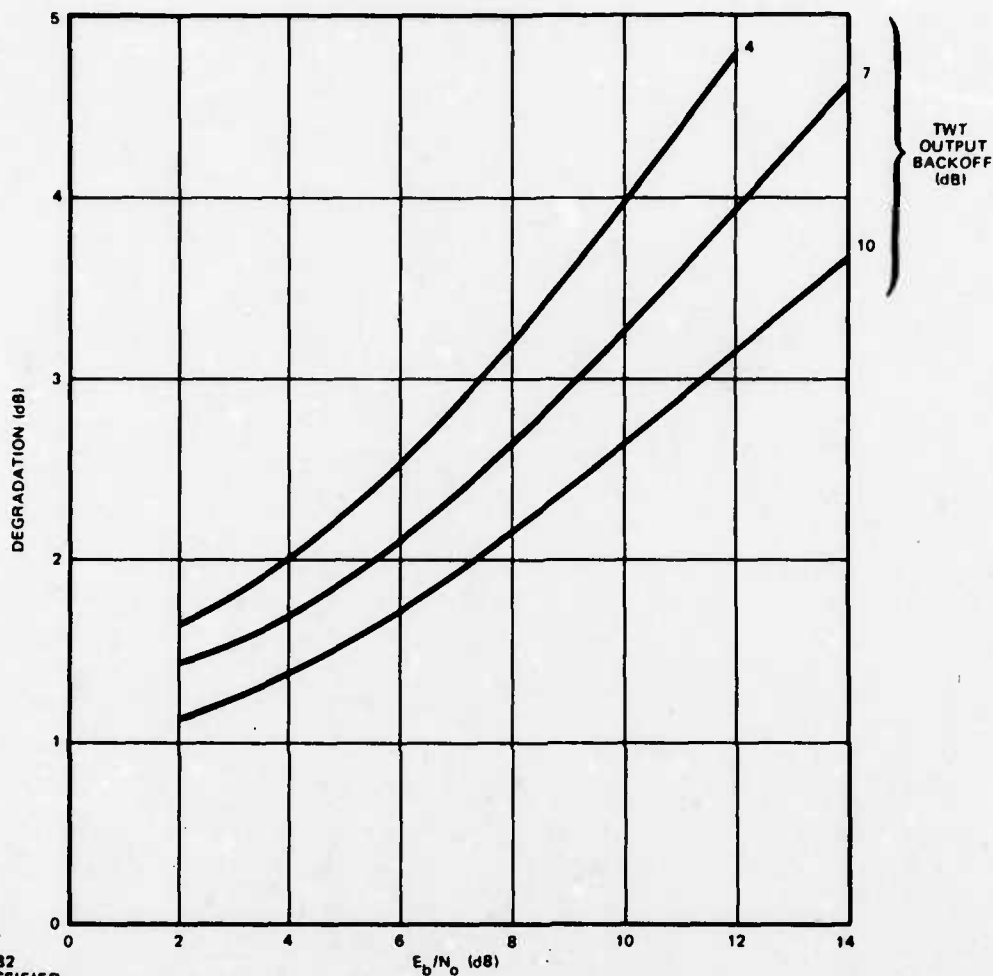
Figure 7-47. Degradation versus Received E_b/N_o (Unequal Signal Levels): Phase II
TWT, Equalized Filters

7.4   CODED FDMA PERFORMANCE IN SATURATING SQPSK CHANNELS

Section 7.3 investigated the degradation to uncoded FDMA in saturating channels. Significant degradation can result with uncoded operation due to the inter-modulation produced by TWT amplifiers. This intermodulation can produce a "floor" so that low error rates can be achieved (if at all) only by a significant increase in receiver E_b/N_o. This section evaluates the degradation that results when the FDMA signals are rate-1/2 convolutionally encoded and Viterbi decoded.

This section assumes equalized filters at the transmitter and receiver and emulation of the ground mobile forces FDMA scenario. Thus five equally-spaced signals access the channel in this study, and probability of bit error is measured by
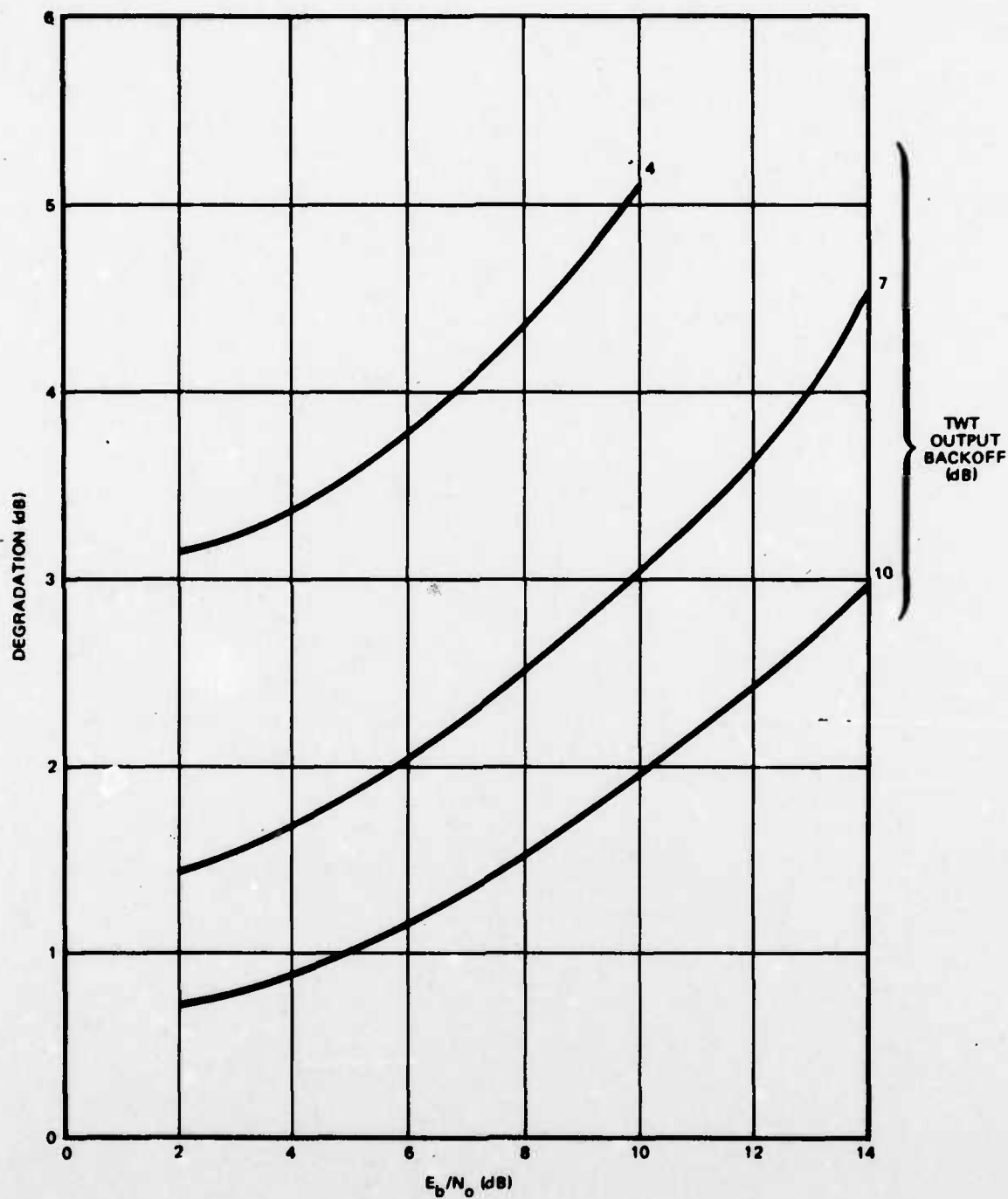
Figure 7-48. Degradation versus Received E_b/N_o (Unequal Signal Levels): Phase III TWT, Butterworth Filters (BT_b = .75)
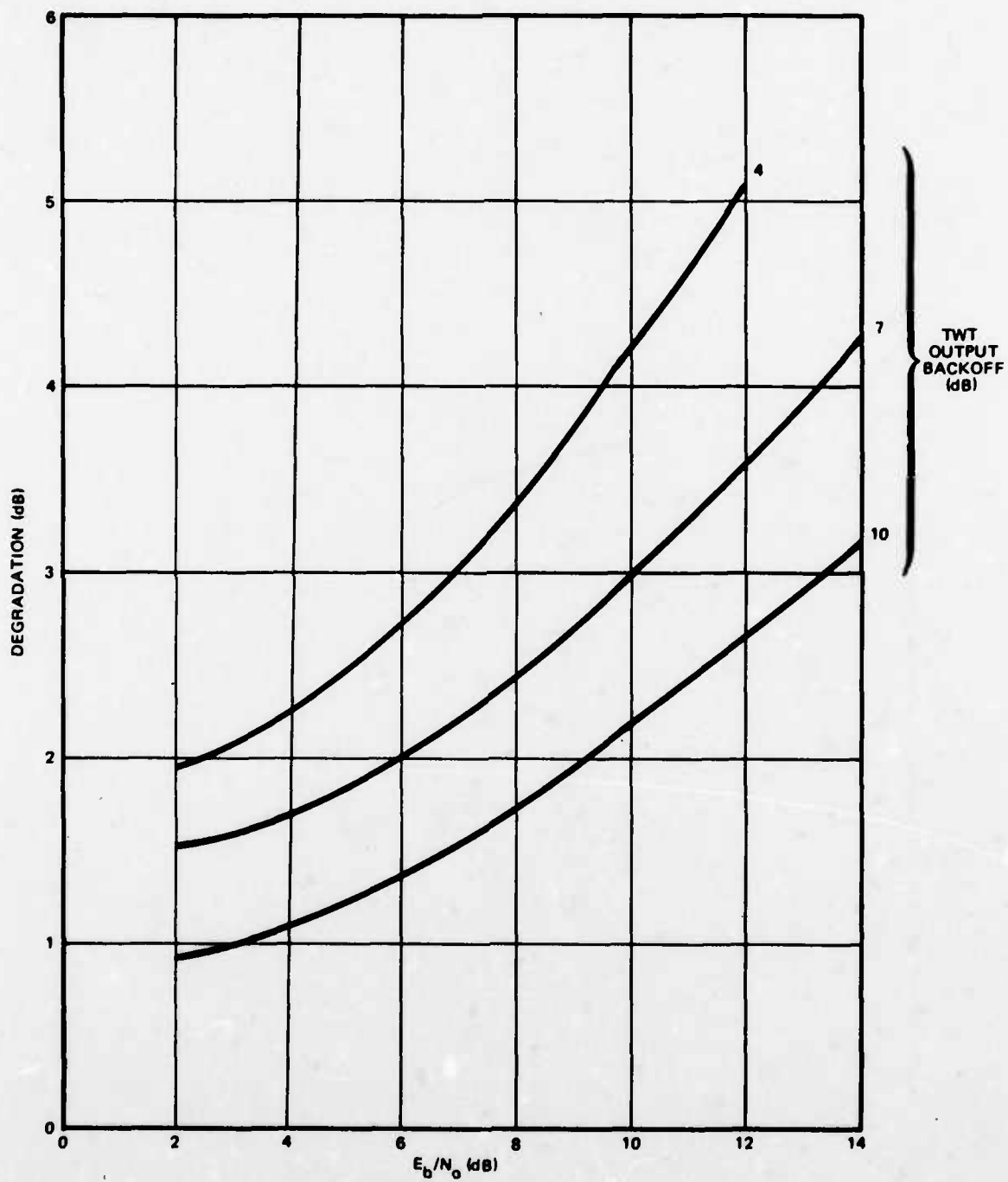
Figure 7-49. Degradation versus Received E_b/N_o (Unequal Signal Levels): Phase III
TWT, Equalized Filters

Viterbi decoding the center channel. Two cases are assumed for signal levels. The desired mode has all accessing signals of equal power while the worst case condition has the center signal at 6 dB below the levels of the adjacent signals. The Phase II TWT model is used.

System degradation is measured by first determining the input E_b/N_o which would produce the same resulting probability of bit error in an ideal system using a convolutional encoder/Viterbi decoder with the same parameters. This (ideal) E_b/N_o is converted to decibels and subtracted from the actual receiver input E_b/N_o (in dB) used in the simulation run.

The following coding parameters were used:

Coding Mode = Serial
Rate (R) = 1/2
Constraint Length (K) = 5
Path Delay (M) = 3 · K
Modulation Format = Staggered Quadriphase (SQPSK)

The number of decoded bits run through the Monte Carlo simulation produced at least ten bit errors. This entailed running several hundred thousand decoded bits at the lower error rates considered. Received E_b/N_o varied from 2 dB to 5 dB, covering the usual range in which coding is performed. Figure 7-50 includes the curve of probability of bit error versus E_b/N_o assuming an ideal SQPSK channel. The results are in agreement with theoretical predictions.

Three signal spacings were selected based on previous results using equalized filters and a Phase II TWT in the satellite. These results showed that degradation was relatively constant for FCENT > 1.5, reached a minimum at FCENT ≐ 1.3, and increased below that point. A spacing less than FCENT = 1.1 is not deemed feasible. Therefore, all curves were evaluated at these three values of signal separation.

Figure 7-50 shows the probability of bit error as a function of received E_b/N_o and signal spacing at a TWT output backoff of approximately 2 dB. Results in 7.3.2 showed that total system loss was minimized for a backoff between 2 and 3 dB for the Phase II TWT and uncoded operation. All five signals are of equal power.

CIRCLES GIVE ERROR RATES AT
E_b/N_o = 5 dB WITH HARD LIMITERS
IN TRANSMITTER AND TWT OUTPUT
BACKOFF = 2 dB

FCENT = 1.1
FCENT = 1.5
FCENT = 1.3

• 1.1
• 1.3
• 1.5

IDEAL CODED CHANNEL

PROBABILITY OF ERROR

E_b/N_o (dB)
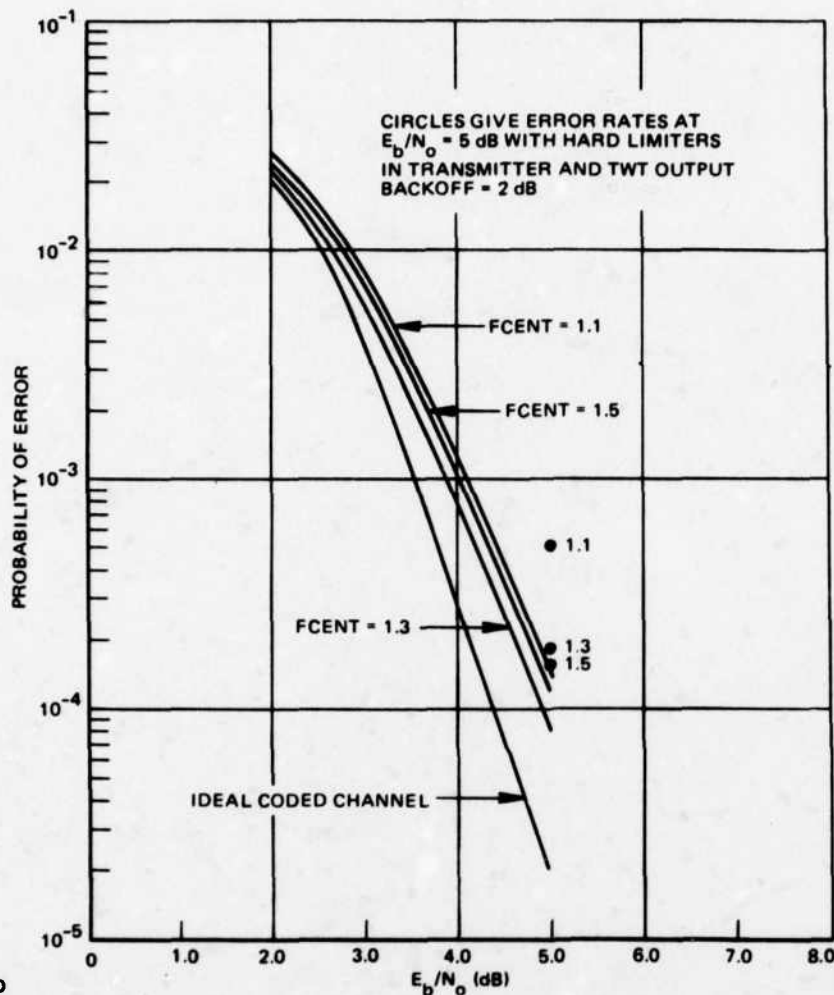
776-4003
UNCLASSIFIED

Figure 7-50. Probability of Error Versus E_b/N_o as a Function of Signal
Separation for TWT Output Backoff = 2 dB (Code Rate = 1/2,
Equal Signal Levels)

At this backoff, for a probability of bit error = 10^{-5} for uncoded operation, results in
Section 7.3.2 showed about a 4 dB degradation at FCENT = 1.5. Figure 7-50 indicates
that the coded loss would be under 1 dB at the same backoff and error rate, compared
to the ideal curve, for coded operation. Uncoded operation at E_b/N_o = 2 dB and out-
put backoff = 2 dB gave a degradation at FCENT = 1.5 of about 0.7 dB. This has been
hypothesized to predict the comparable coded degradation and does give close agree-
ment for the cases considered. The coded degradation at FCENT = 1.1 is about 0.1
dB greater than at FCENT = 1.5 while the degradation at FCENT = 1.3 is about 0.2
dB less than that at a spacing of FCENT = 1.5.

7-60

The circles on Figure 7-50 indicate the corresponding performance at E_b/N_o = 5 dB when hard limiters are inserted in the transmitters. Again, the Phase II satellite TWT is operated at an output backoff of 2 dB.

Figure 7-51 plots probability of bit error versus TWT output backoff as a function of signal spacing assuming a received E_b/N_o = 5 dB and unequal signal levels. This received E_b/N_o produces a probability of bit error = 2×10^{-5} in the ideal channel. In this simulation, all four adjacent signals are 6 dB stronger than the center channel in which the probability of bit error is computed.
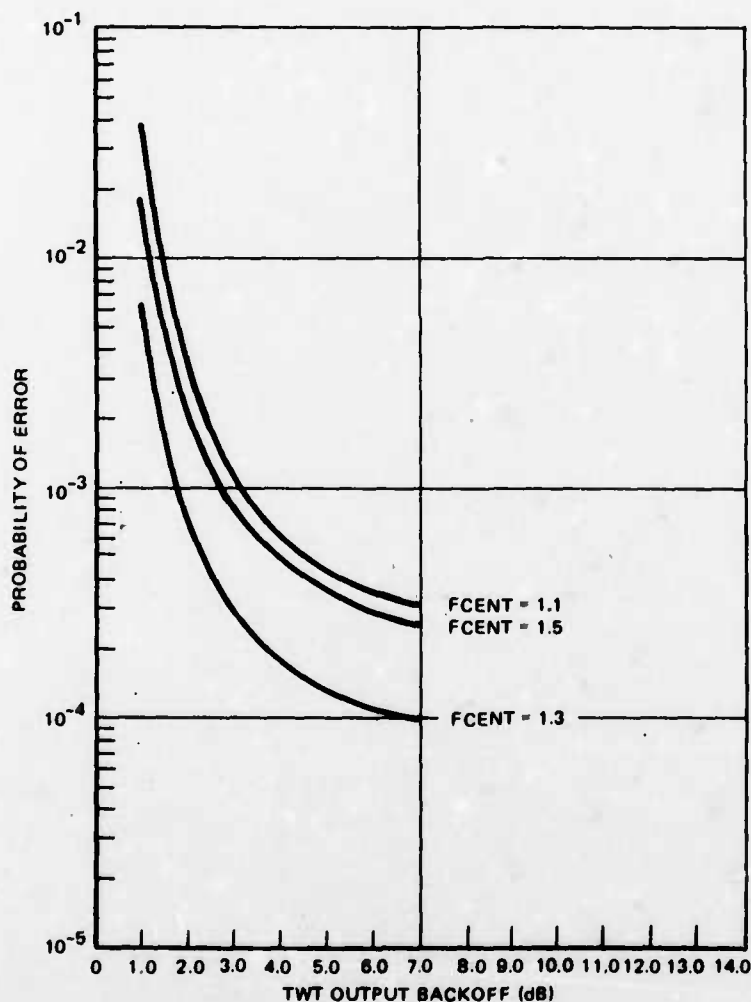
776-4004
UNCLASSIFIED

Figure 7-51. Probability of Error Versus TWT Output Backoff as a Function of Signal Spacing for E_b/N_o = 5 dB (4 Adjacent Signals Each 6 dB Stronger than Coded Center Signal)

7-61

Curves in Figure 7-51 show that very large degradation results from this power differential, especially for low values of TWT backoff. For uncoded data, at a TWT backoff = 2 dB and FCENT = 1.5, Figure 7-36 shows that a degradation of over 2 dB results at E_b/N_o = 2 dB and an unmeasurable degradation occurs at E_b/N_o = 10 dB. With coding, the corresponding degradation is 1.7 dB at E_b/N_o = 5 dB. Again, the uncoded degradation at E_b/N_o = 2 dB is a fairly good estimate of coded performance.

These Viterbi runs illustrate several important results. First, the Monte Carlo simulation results indicate that the degradation is close to the value predicted by the degradation at E_b/N_o = 2 dB for an uncoded system. This uncoded estimate can be obtained far more easily and faster than a full Monte Carlo simulation.

Coding produces much less degradation due to the effects of saturation and AM-PM conversion than occur in the corresponding uncoded system. In addition, degradation is comparable at a spacing FCENT = 1.1 as for FCENT $\geq$ 1.5. Degradation at FCENT = 1.3 is somewhat less for equal signal levels with the difference in degradation increasing as the ratio of power in the adjacent channels to that in the center channel increases.

Note that while degradation for a fixed error rate is much less for the coded system at a given null-to-null spacing (FCENT), the actual channel separation used for the coded channel is twice that used for the equivalent uncoded channel for rate-1/2 coding. Also, a decrease in error rate would result for an increase in coder constraint length (at the expense of additional complexity in the decoder).

The basic conclusion is that coding improves the performance of saturating channels and is highly desirable where low error rates are necessary and the doubling of bandwidth can be tolerated.

SECTION VIII
TDMA PERFORMANCE

This section examines the performance and bandwidth requirements for
TDMA service using QPSK or SQPSK signalling. The effects of filter characteristics,
phase distortion, and saturating amplifiers are determined for several channel
configurations. The cases of both single and multiple hard limiters in the channel are
examined. The use of equalizers in the transmitter or receiver to provide perfor-
mance improvement is investigated.

The results of Section 7.1 showed that use of the phase distortionless equal-
ized filters instead of phase distortionless Butterworth filters produced an increase
in channel capacity of at least 25 percent. The results of that study are applicable to
TDMA operation and indicate that very low degradation results if the TDMA data rate
is twice the 3 dB bandwidth of the equalized filters in the transmitter and receiver.
Both the equalized filters and phase distortionless 5-pole Butterworth filters are
evaluated for TDMA operation.

8.1   EFFECTS OF SATURATING AMPLIFIER ON TDMA

This section describes the degradation due to a saturating amplifier on a
single accessing quadriphase (QPSK) or staggered quadriphase (SQPSK) modulated
signal. Phase distortionless 5-pole equalized filters (Figure 7-3) are used in the
transmitter and receiver. An integrate-and-dump detector is used.

Initially, a linear AM-PM conversion characteristic was used, as shown
in Figure 3-11. A linear amplitude characteristic was assumed. The phase shift
is specified in degrees per decibel above a floor below which no phase shift occurs.
The input signal is adjusted so that the average input power is 10 dB above the floor.
This input drive is sufficient to produce the maximum degradation value. The results
are shown in Figures 8-1 and 8-2. The resulting degradations, measured at received
$E_b/N_o = 2$ dB and $E_b/N_o = 10$ dB, are comparable for QPSK and SQPSK. Degradation
could become severe if the amplifier has a phase slope greater than 8 degrees/dB.
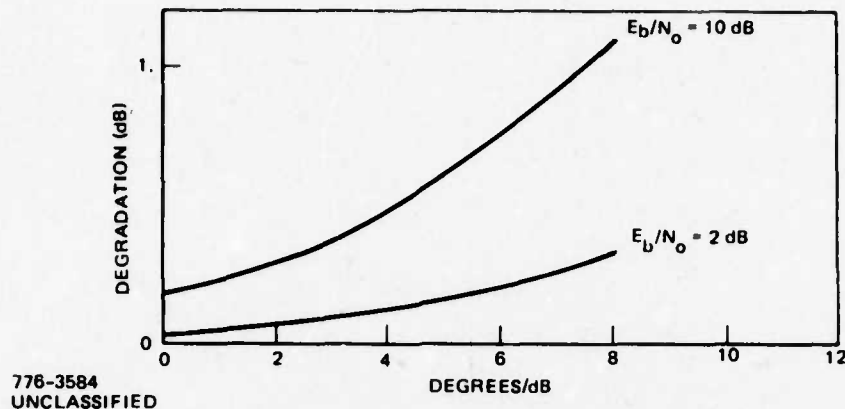
Figure 8-1. Degradation for Single QPSK Signal as a Function of AM/PM Conversion
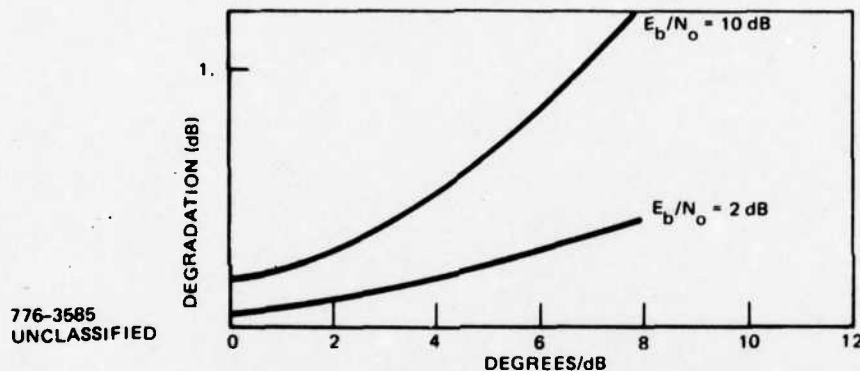
Figure 8-2. Degradation for Single SQPSK Signal as a Function of AM/PM Conversion

Figures 8-3 and 8-4 show the degradation for a single QPSK or SQPSK signal passing through the Phase III TWT. TDMA has been proposed for the DSCS Phase III satellite. Again, both transmit and receive equalized filters are used and an integrate-and-dump detector is assumed. Degradation is plotted as a function of TWT output backoff for received $E_b/N_o = 2$ dB and $E_b/N_o = 10$ dB. No major differences in degradation exist for QPSK and SQPSK. Since, as backoff is decreased, the increase in degradation is always less than the corresponding increase in output power, no optimum backoff level exists for single access operation. No turnaround is exhibited and as a result the TWT should be operated at saturation to minimize the total system loss.

8.2 SQPSK TRANSMISSION IN A HARD LIMITING CHANNEL

This section examines the bandwidth requirement for transmitting SQPSK through a hard limiting channel, such as would be employed for TDMA. The transmitter is narrowband filtered prior to hard limiting, * and the receiver incorporates

*The limiter is in either the satellite channel or the terminal transmitter, but not both for this discussion.

776-3586
UNCLASSIFIED
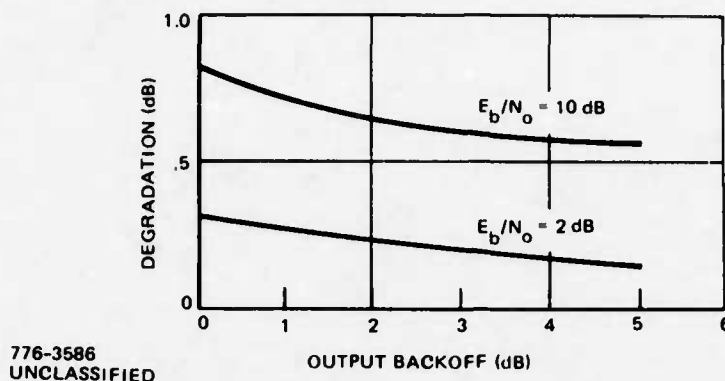
Figure 8-3. Degradation of Single QPSK Signal With Phase III TWT Backoff

776-3587
UNCLASSIFIED
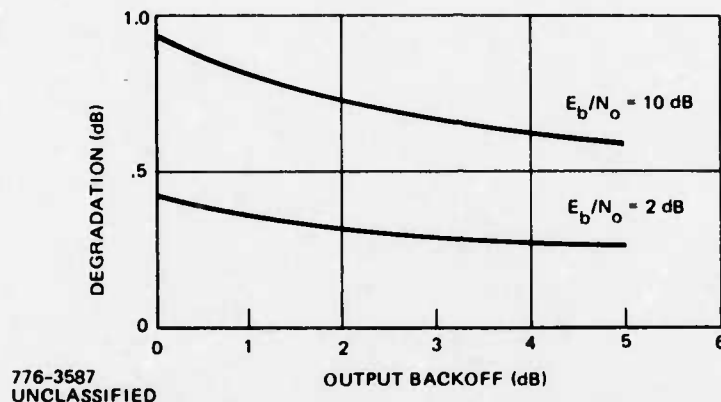
Figure 8-4. Degradation of Single SQPSK Signal With Phase III TWT Backoff

an integrate-and-dump filter. The receiver can be wideband or narrowband filtered and adaptively equalized using the technique described in 5.2.2.1. Zero phase filters with a Butterworth amplitude characteristic are assumed. *

The effect of phase distortion introduced prior to the hard limiting is of particular concern, since in Section 6.2 it was shown that receiver equalization is not fully effective for correcting such distortion. Consequently, if such phase distortion is excessive, an equalizer is needed in the transmitter prior to the hard limiting. A practical technique for adjusting the transmitter's equalizer must be devised.

8.2.1   BANDWIDTH REQUIREMENT FOR SQPSK IN HARD LIMITING CHANNEL

The study of QPSK or SQPSK for the linear channel in 7.1.2 shows that improved bandwidth utilization for PSK with integrate-and-dump detection is achieved by adaptive equalization, which tends to boost the amplitude response at the edge of the passband. We now study the hard limiting channel.

---

*The numerical results herein are slightly in error because the filter symmetry is offset by one frequency increment, or 3 percent for 32 samples per bit. It was not deemed worthwhile repeating the computations to correct this error.

As the simplest system model of SQPSK in a hard limiting channel, we assume a narrowband filter on the transmitter prior to the hard limiting and a wideband receiver with integrate-and-dump detection. The filter has a 5-pole Butterworth amplitude response and zero phase shift. The degradation without equalization in either the transmitter or the receiver is given in Table 8-I for two bandwidths (along with many other results yet to be described). Performance in the hard limiting channel is about the same as in the nonsaturating channel for the same filters, as given in Figure 7-2.

Now, a decision-directed adaptive equalizer, controlled by the LMS algorithm, is introduced in the receiver, the objective being to attempt to compensate for the narrow-bandpass prior to the hard limiting. Table 8-I shows that essentially no improvement results from the receive equalization, in contrast to the behavior in a nonsaturating channel, as given in Figure 7-2.

As the next step of system complexity, a receive filter is utilized with the same bandwidth as the transmit filter. Without equalization, performance is slightly degraded, compared to the wideband receiver. Allowing the receive equalizer to adapt, some improvement in performance is achieved. In contrast, in the nonsaturating channel, the degradation from ideal QPSK is reduced almost to zero for the same filters and a receive equalizer, (Figure 7-2).

8.2.2   EFFECT OF TRANSMITTER EQUALIZATION ON BANDWIDTH
         UTILIZATION

The effect of equalization at the transmitter prior to the hard limiting is now considered for a narrow bandwidth transmit filter. Special treatment of this case is necessary when developing a computer simulation. Adaptive equalization in the receiver is much easier to simulate because superposition can be applied to compute the consequence of a change in a tap weight. Superposition is not valid in a saturating system.

The criterion employed for adjusting the transmit equalizer is minimization of mean square error in the bit decision samples (soft decisions) out of the receiver. For the simplest system model, the receiver is wideband with integrate-and-dump detection to obtain the bit decision samples. Since the transfer function of the receiver remains fixed, noise can be ignored in computing the mean square error as a function of the tap weights of the transmit equalizer; i.e., the error is due entirely to the intersymbol interference.

Table 8-I. SQPSK In Hard Limiting Channel

Channel Condition	Equalization Transmit	Receive	Transmit Bandpass	Receive Bandpass	Degradation at $E_b/N_o = 10$ dB
No Phase Distortion	No	No	$1.0/T_b$	∞	2.2 dB
No Phase Distortion	No	No	$1.2/T_b$	∞	1.5 dB
No Phase Distortion	No	Yes	$1.0/T_b$	∞	2.1 dB
No Phase Distortion	No	Yes	$1.2/T_b$	∞	1.3 dB
No Phase Distortion	No	No	$1.0/T_b$	$1.0/T_b$	2.9 dB
No Phase Distortion	No	No	$1.2/T_b$	$1.2/T_b$	1.6 dB
No Phase Distortion	No	Yes	$1.0/T_b$	$1.0/T_b$	1.4 dB
No Phase Distortion	No	Yes	$1.2/T_b$	$1.2/T_b$	0.9 dB
No Phase Distortion	Yes	No	$1.0/T_b$	∞	1.5 dB
No Phase Distortion	Yes	Yes	$1.0/T_b$	∞	1.8 dB
No Phase Distortion	Yes	No	$1.2/T_b$	∞	0.8 dB
No Phase Distortion	Yes	Yes	$1.2/T_b$	∞	1.1 dB
No Phase Distortion	Yes	No	$1.2/T_b$	$1.2/T_b$	0.7 dB
No Phase Distortion	Yes	Yes	$1.2/T_b$	$1.2/T_b$	0.7 dB
$1 - j\,.6\,\sin(1.25\,\omega T_b)$	No	No	$1.2/T_b$	∞	4.2 dB
$1 - j\,.6\,\sin(1.25\,\omega T_b)$	No	Yes	$1.2/T_b$	∞	1.7 dB
$1 - j\,.6\,\sin(1.25\,\omega T_b)$	Yes	No	$1.2/T_b$	∞	1.0 dB
$1 - j\,.6\,\sin(1.25\,\omega T_b$	No	Yes	$1.2/T_b$	$1.2/T_b$	1.2 dB
$1 - j\,\sin(1.25\,\omega T_b)$	No	No	$1.2/T_b$	∞	Very Large
$1 - j\,\sin(1.25\,\omega T_b)$	No	Yes	$1.2/T_b$	∞	2.9 dB
$1 - j\,\sin(1.25\,\omega T_b)$	Yes	No	$1.2/T_b$	∞	1.3 dB

5-Pole Butterworth Amplitude, Zero Phase

5-Tap Transmit Equalizer, 8-Tap Receive Equalizer

T_b = Bit Duration on Inphase or Quadrature

Tap Spacing = T_b

Integrate-and-Dump Detector

The ideal SQPSK signal is filtered and passed through the transmit equalizer. The filtered and equalized signal is then hard limited and input to the integrate-and-dump detector in the receiver. The receiver also extracts a clock tracking error

voltage from the bit transitions on the inphase component. The closed loop tracking drives the clock tracking error to zero to optimize the dump times for integrate-and-dump detection. (With zero phase filters and no phase distortion in the channel, the dump time should be identical with the unfiltered transitions.)

The adjustment procedure for the transmit equalizer is basically to perturb each tap weight and observe the corresponding change in the mean square error in the bit decision samples over one period of the signal. Then, the tap weights are adjusted in the direction to reduce the mean square error. For the present study, only complex symmetric channels are considered (amplitude response is an even function of frequency and phase shift is an odd function); hence, the tap weights of the transmit equalizer can be constrained to be real, reducing the number of variables by a factor of two. Furthermore, the center tap weight can be fixed equal to unity in view of the fact that the hard limiter output is independent of the absolute gain through the equalizer. However, the mean square error on the bit decision samples is dependent on the gain after the hard limiter (or, equivalently, on the amplitude assigned to the hard decisions). This post-limiter gain factor is adjusted together with the tap weights (other than the center tap weight) to minimize the error. Three versions of the adjustment procedure were tried, and all were successful for the examples studied.

The first version utilized an available FORTRAN subroutine which searches to find the minimum of a function of several real variables without requiring evaluation of the gradient*. In this application, the function is the mean square error, and the real tap weights are the variables. This routine successfully adapted a 5-tap equalizer; however, it has the disadvantage of tending to continue the search procedure well beyond the accuracy actually needed for equalization. The subroutine is supposed to terminate the search when the tap weights do not change within the specified accuracy, but it tends to go beyond this point.

The second version used a sequential step search. A tap weight is incremented either up or down by a specified step size in an attempt to reduce the mean square error. The selected tap weight is incremented with the step size fixed until the error no longer reduces. Then, the procedure is repeated on the next tap

---

*SUBROUTINE CDMIN, developed at University of Waterloo to minimize a function of several variables, does not require evaluation of derivatives of the function with respect to the variables. In Appendix E, this subroutine is applied to the problem of adjusting an equalizer to minimize probability of error, instead of minimizing mean square error.

weight. This search procedure is similar to a manual adjustment procedure. The procedure is terminated when the error can no longer be reduced for a step size corresponding to the desired accuracy.

The third version adjusts the tap weights sequentially by a dither scheme, which computes a finite difference approximation to the gradient of the error. A tap weight is first increased and then decreased by the specified step size. The difference in the two resulting errors is used to proportionately increment the tap weight in the direction to reduce the error. Then, the next tap weight is similarly controlled. The proportionality factor must be small enough that the control is stable and the solution is convergent. The procedure is terminated when the tap weights appear to have converged.

Transmit equalization was derived for the two transmit filter bandwidths previously assumed with a wideband receiver. The resulting degradations are given in Table 8-I. The transmit equalizer has improved performance to some extent. Then, the receive equalizer was allowed to adapt, with the interesting result that the degradation actually increased slightly. Finally, a receive narrowband filter was introduced. It has almost no effect on performance, even when the receive equalizer is allowed to adapt.

The degradation for hard limited SQPSK is reasonably small when the passband is $1.2/T_b$. There is only a small benefit from equalization at either end of the link, in the absence of phase distortion in the channel. The best performance achieved here is essentially the same as found in Section 7.1.2.4 using identical, equalized filters at both ends of the link, where the equalization is derived for a nonsaturating channel.

8.2.3   EQUALIZATION OF TRANSMITTER PHASE DISTORTION

We now study equalization of phase distortion in the transmitter. As a fairly severe test, the distortion transfer function is assumed to be $1 - j \, .6 \sin(1.25 \, \omega T_b)$, which has ± 0.7 dB of variation in amplitude and $\pm 31^o$ of phase distortion. Unequalized, the performance degradation is excessive (4.2 dB) as a consequence of the intersymbol interference due to the distortion.

Equalization only in the receiver gives a significant performance improvement, but there still is some degradation from the performance achieved in the absence of the distortion (but for otherwise identical channel parameters). Equalization in the transmitter yields more improvement than equalization in the receiver,

almost approaching the case of no distortion, but the difference is not large (1.0 dB degradation for transmit equalization compared with 1.7 dB for receive equalization).

Next, an extreme distortion of the form $1 - j \sin(1.25 \omega T_b)$ was assumed, which has ± 1.5 dB of variation in amplitude and $\pm 45^o$ of phase distortion. Unequalized, the degradation is very great because the intersymbol interference is on the verge of causing errors even in the absence of noise.

Equalization only in the receiver yields a significant improvement, but the resulting degradation is still excessive. Equalization in the transmitter provides considerably more performance improvement (degradation is 1.3 dB for transmit equalization versus 2.9 dB for receive equalization). There still remains 0.5 dB degradation compared with the case of no distortion (for otherwise identical channel parameters); possibly, this is a consequence of assuming an insufficient number of taps for the transmit equalizer.

8.2.4   CONTROL OF TRANSMIT EQUALIZER

We now raise the question of how to derive control of the transmit equalizer when the signal accessible after the phase distortion is hard limited. Receiving the signal locally at the terminal enables the transmit equalizer to be controlled from the decision-directed mean square error measured in the receive modem, much like the computer simulation approach. The tap weights of the transmit equalizer are adjusted (manually or automatically by a sequential dither scheme) to minimize the error in the receiver. It is assumed that the receiver does not have significant phase distortion or has already been equalized.

An interesting question, not settled by the computer simulation reported herein, is whether independent equalizers in the transmitter and receiver can be controlled simultaneously. Since the objective for both is minimizing the mean square error, there presumably is no difficulty.

If the signal is received from the satellite repeater, which imposes the hard limiting, the equalizer control must take into consideration the propagation lag (1/4 second for a stationary satellite). This necessitates a very narrow bandwidth for an automatic control loop to avoid instability. With a dither control scheme, the propagation lag must be taken into account when comparing the error for, respectively, an increase and a decrease in the tap weight.

8.2.5 CONCLUSIONS

If the SQPSK channel is hard limited following a narrow transmit filter with no phase distortion, there is some benefit from equalization either in the transmitter or the receiver. Somewhat more benefit is realized from transmit equalization than from receive equalization, but the difference is relatively small, particularly when the receiver also has a narrow filter*. Note that the distinction between transmit equalization and receive equalization to correct phase distortion is not meaningful in a nonsaturating channel.

If phase distortion is introduced in the transmitter prior to hard limiting, a receive equalizer provides a substantial performance improvement relative to unequalized operation. A transmit equalizer yields more improvement but the difference becomes significant only when the phase distortion becomes very severe.

If the phase distortion prior to hard limiting is not too large, say less than $\pm 30^{\circ}$ for sinusoidal type, the additional improvement available from transmit equalization, compared with receive equalization, is relatively small.

It is expected that phase distortion for SQPSK/TDMA operation with a hard limiting transmitter or satellite will conform to the above restriction. Thus, a receive equalizer should suffice to correct phase distortion in the transmitter. There still is the operational consideration with TDMA that only one receiver is implemented to receive all accesses via time-sharing. It is not clear how to control a time-shared receive adaptive equalizer to correct different distortions in the various transmitters. One possibility is to provide computer control and store the tap weights digitally. At the onset of each TDMA burst, the tap weights are inserted into the receive equalizer, and adaptation continues for the burst. Then, the adapted tap weights are returned to digital storage until a TDMA burst from that transmitter is again scheduled.

---

*Present TDMA implementations use integrate-and-dump detectors. Integrate-and-dump detection of SQPSK is therefore presumed for both wideband and narrowband receive filters, although it may be a disadvantage in the narrowband receiver.

8.3 TDMA WITH TRANSMIT HARD LIMITING AND SATELLITE HARD LIMITING

This section addresses the topic of TDMA for DSCS digital communication by postulating saturation in the transmitter as well as in the satellite channel. To take advantage of TDMA by maximizing downlink power and avoiding need for uplink power control of the accesses, it is convenient to have the terminal transmitter drive the satellite channel into hard limiting. Particularly to get the requisite uplink EIRP out of smaller terminals to saturate the satellite, TDMA may also imply operation of the terminal transmitter in hard limiting. Therefore, cascading of two hard limiters, each preceded by a narrow filter, is assumed for the present discussion.

QPSK or SQPSK modulation is assumed, and performance is computed at $E_b/N_o = 10$ dB, corresponding to uncoded data. Degradation due to intersymbol interference is typically less at the lower E_b/N_o needed with error correction coding.

Because the emphasis in this discussion is placed on bandwidth utilization, phase distortionless (i.e., equalized) 5-pole Butterworth filters are assumed in the transmitter and receiver. However, distortion can be introduced in the transmitter, if desired. An adaptive equalizer is incorporated in the receiver, controlled from the bit decision samples in the receiver output. A transmit equalizer can also be used prior to the hard limiting. In contrast to Sections 8.1 and 8.2, the receiver utilizes a sampling detector rather than an integrate-and-dump detector, since the latter tends to cause more degradation with a narrowband receiver.* (With a receive equalizer, the presence of the integrate-and-dump makes no difference because the equalizer's response can compensate, provided the response bandwidth is within the spectral nulls.)

8.3.1 QPSK VERSUS SQPSK

In a linear channel with a symmetrical transfer function, QPSK and SQPSK have exactly the same performance since there is no interaction between the inphase and the quadrature data streams. In a hard limiting channel, this is no longer the case. Since SQPSK has offset transitions, there is an envelope

---

*Note that without intersymbol interference, a receiver with the same phase distortionless filter as the transmitter and an integrate-and-dump detector would maximize signal-to-noise ratio.

fluctuation depending on whether a transition actually takes place. The action of the hard limiter boosts the inphase amplitude when a quadrature transition occurs, and vice versa.

A major characteristic of SQPSK, compared with QPSK, is that the out-of-band spectrum is not restored by hard limiting after sharp cutoff filtering. This behavior of SQPSK to retain low out-of-band spectral emissions is useful for FDMA operation when the transmitters can saturate. It is not of significance for TDMA where there is only one access at a time*.

The selection between QPSK and SQPSK for TDMA operation depends on attainable performance in a narrowband saturating channel. Section 8.1 showed a slight advantage for QPSK with AM-PM conversion and the Phase III satellite TWT.

8.3.2   BANDWIDTH REQUIREMENT FOR TDMA

To begin, we assume the same bandwidth in transmitter, satellite channel, and receiver. Then the transmitter bandwidth or the satellite channel bandwidth is increased. Both SQPSK and QPSK are treated   and there is no phase distortion. Results are presented in Table 8-II. It is observed that QPSK consistently has less degradation than SQPSK for the same bandwidth filters. The advantage of QPSK over SQPSK is in the range 0.5 dB to 1.0 dB. Thus, TDMA operation prefers QPSK.

Allowing a receive equalizer to adapt in an attempt to improve performance with a sampling detector yields no significant benefit, and may even increase degradation in some cases.

Another possibility is a transmit equalizer. The approach described in Section 8.2.2 using the receiver bit decision amplitudes to adjust the tap weights of a transmit equalizer was applied to the present case of cascaded hard limiters. When there is no phase distortion, the transmit equalizer produces no improvement; hence, it may be concluded that the maximally flat Butterworth amplitude characteristic of the filter is essentially optimum for a receiver with a sampling detector.**

---

*This statement presumes no problem of adjacent channel interference in a channelized satellite repeater.

**Section 8.2.1 showed a benefit from either receive or transmit equalization when the receiver has an integrate-and-dump detector.

Table 8-II. QPSK and SQPSK Performance With Both Transmitter Hard Limiting and Satellite Channel Hard Limiting

Channel Condition	Equalizer	Transmit Bandpass	Satellite Bandpass	Receive Bandpass	Modulation	Degradation at E_b/N_o = 10 dB
No Phase Distortion	No	$1.0/T_b$	$1.0/T_b$	$1.0/T_b$	SQPSK	3.6 dB
"	Receive	$1.0/T_b$	$1.0/T_b$	$1.0/T_b$	SQPSK	3.2 dB
"	No	$2.0/T_b$	$1.0/T_b$	$1.0/T_b$	SQPSK	1.9 dB
"	No	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	1.9 dB
"	Receive	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	2.1 dB
"	No	$2.4/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	1.2 dB
"	No	$1.0/T_b$	$1.0/T_b$	$1.0/T_b$	QPSK	2.6 dB
"	Receive	$1.0/T_b$	$1.0/T_b$	$1.0/T_b$	QPSK	2.3 dB
"	No	$2.0/T_b$	$1.0/T_b$	$1.0/T_b$	QPSK	1.5 dB
"	No	$2.0/T_b$	$2.0/T_b$	$1.0/T_b$	QPSK	1.2 dB
"	Receive	$1.0/T_b$	$2.0/T_b$	$1.0/T_b$	QPSK	1.6 dB
"	No	$1.0/T_b$	$1.0/T_b$	$1.2/T_b$	QPSK	1.3 dB
"	No	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	1.2 dB
"	Receive	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	1.2 dB
"	No	$2.4/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	0.7 dB
"	No	$2.4/T_b$	$2.4/T_b$	$1.2/T_b$	QPSK	0.7 dB
"	Receive	$1.2/T_b$	$2.4/T_b$	$1.2/T_b$	QPSK	0.8 dB
"	Transmit	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	1.2 dB
$1 - j.6 \sin(1.25 \, \omega T_b)$ In transmitter	No	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	5.0 dB
"	Receive	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	2.1 dB
"	No	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	4.5 dB
"	Receive	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	2.9 dB
"	Transmit	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	1.3 dB
$1 - j.6 \sin(1.25 \, \omega T_b)$ In satellite	Receive	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	2.0 dB

5-Pole Butterworth amplitude characteristic, zero phase shift filters
Sampling Detector
5-Tap transmit equalizer; 8-Tap receive equalizer
T_b = Bit duration on inphase or quadrature
Tap Spacing = T_b

The results of Table 8-II show feasibility of TDMA operation with a bandpass for all filters (e.g., transmitter, satellite channel, and receiver) as narrow as 1.2 times the bit rate on inphase or quadrature, and allowing more degradation, even narrower. Thus, for the 50 MHz bandpass available at the 70 MHz IF of the COMTECH up and downconverters (see Section 2.2), a burst data rate of 80 Mbps, or 40 Mbps with rate-1/2 coding, can be transmitted. There is some benefit from using a wider transmit bandwidth, available from the 700 MHz IF, prior to the transmit hard limiting. Also, there is benefit from a wider satellite channel bandwidth.

8.3.3 PHASE DISTORTION

We now examine the problem of equalizing phase distortion in the transmitter or in the satellite. For SQPSK, the capability of a receive equalizer to correct phase distortion prior to hard limiting in the satellite has already been demonstrated in Section 8.2.3. Here, we treat the case of distortion prior to hard limiting in the transmitter, with hard limiting also in the satellite channel. The distortion transfer function is sinusoidal type, with ± 0.7 dB of amplitude variation and $\pm 31^{\circ}$ of phase distortion. Without an equalizer to correct the distortion, the degradation for SQPSK is seen in Table 8-II to be large (5.0 dB); however, a receive equalizer brings the SQPSK performance almost to the no distortion value for the same filters.

We now consider QPSK. Introducing the distortion in the transmitter prior to hard limiting again produces a large degradation (4.5 dB), as seen in Table 8-II. Allowing a receive equalizer to adapt, it is found that performance is improved, but the degradation is still excessive (2.9 dB compared with 1.2 dB in the absence of the transmitter distortion).

A transmit equalizer was now introduced and adjusted from the receiver's bit decision amplitudes. As seen in Table 8-II, the transmit equalizer essentially fully corrects the distortion occuring in the transmitter. This is no surprise, since the equalizer and the distortion are not separated by the channel nonlinearity for this case.

If the distortion is introduced in the satellite channel, neither a receive equalizer or a transmit equalizer can fully correct the distortion, because of the cascaded hard limiting. With SQPSK, the receive equalizer comes close to full correction as shown previously. With QPSK, a receive equalizer does provide considerable improvement for distortion in the satellite, but still leaves some impairment (less than when the distortion is in the transmitter).

8.3.4   CONCLUSIONS

TDMA operation poses the problem of cascaded hard limiting due to saturation in the terminal transmitter as well as in the satellite channel. Using either SQPSK or QPSK, the filter bandpass can be as narrow as 1.2 times the bit rate (on inphase or quadrature), assuming phase distortionless (i.e., equalized) filters. QPSK outperforms SQPSK and can tolerate a bandpass as narrow as the bit rate, with more degradation. Presuming a sampling detector in the receiver (which is preferred over an integrate-and-dump detector when the receiver is narrowband), the maximally-flat amplitude characteristic assumed for the filters is optimum.

The choice between QPSK and SQPSK is clouded by the effect of phase distortion prior to hard limiting. With SQPSK a receive equalizer controlled from the bit decision samples can correct fairly severe distortion (e.g., $\pm30^{o}$ ripple type) occuring in the transmitter. With QPSK a receive equalizer only partially corrects the distortion, and excessive degradation remains. A transmit equalizer is, therefore, necessary with QPSK to correct distortion occuring in the transmitter prior to the hard limiting.

Neither a transmit equalizer nor a receive equalizer can fully correct distortion occuring in the satellite channel. This points out the desirability of insuring good filter equalization in the satellite channels.

8.4   QPSK AND SQPSK PERFORMANCE WITH MULTIPLE CASCADED NARROWBAND HARD LIMITERS

TDMA operation through a satellite channel will typically drive the satellite repeater into saturation, thus maximizing downlink power and eliminating need for accurate uplink power control. A large terminal with high EIRP can saturate the satellite with its transmitter at reduced power, thereby staying linear, if desired. However, to saturate the satellite, a smaller terminal may have to drive its own transmitter into saturation. Furthermore, as evidenced by the TSC-86 terminal, the transmitter design may include limiting in the upconverter as well.

To obtain a feel for the deleterious effect of multiple saturating amplifiers of finite bandwidth, QPSK and SQPSK performance through cascaded bandpass hard limiters is studied. For this discussion, phase distortionless filters are presumed with a 5-pole Butterworth amplitude cutoff characteristic. The receiver incorporates a filter and sampling detector, and, if desired, an adaptive equalizer controlled by the LMS algorithm acting on the bit decision samples.

To begin, we assume a wideband terminal transmitter; this will be the reference for comparison when additional filters and limiters are introduced. The performance of SQPSK and QPSK in a relatively narrowband hard limiting satellite channel is presented in Table 8-III. It was determined that QPSK somewhat outperforms SQPSK when the channel contains hard limiting. (There is no difference in a linear channel with symmetrical filters.) For the assumed phase distortionless filters, there is no benefit from receive equalization when the receiver has the same bandwidth as the channel, and a sampling detector is employed.

A receive equalizer has been shown to improve performance considerably when the receiver has an integrate-and-dump detector instead of a sampling detector. Furthermore, after equalization, performance with integrate-and-dump is slightly better than observed here for a sampling detector. This is explained as due to reduction of noise outside the nominal passband. There should be no difference if the filters had sharp cutoff outside the passband $-.5/T_b < f < .5/T_b$, since the equalizer can create an arbitrary transfer function within this passband.

As the next step, the terminal transmitter was made narrowband and hard limiting (some numerical results taken from Table 8-II are given in Table 8-III). By comparing with the case of a wideband transmitter, it is seen that performance is degraded to some extent. QPSK with the bandpass equal to $1.2/T_b$ has a relatively low degradation (1.2 dB at $E_b/N_o = 10$ dB).

Finally, a narrowband and hard limiting up-converter is introduced in addition to the hard limiting transmitter. The effect is to degrade performance further. However, by widening all filters relative to the data rate, the degradation is reduced. At a bandpass of $1.4/T_b$ for all filters, the performance with the narrow, hard limiting up-converter is about the same as for a bandpass of $1.2/T_b$ and a wideband up-converter.

In conclusion, cascading narrowband hard limiters has the effect of degrading QPSK or SQPSK performance for TDMA operation. In all cases of phase distortionless filters, QPSK provides somewhat better performance (less degradation) than SQPSK. Improved performance is realized by reducing the data rate in the specified bandwidth.

Table 8-III. Performance of QPSK and SQPSK With Cascaded Hard Limiting

Equalizer	Upconverter Bandpass (Hard Limiting)	Transmitter Bandpass (Hard Limiting)	Satellite Bandpass (Hard Limiting)	Receive Bandpass	Modulation	Degradation at $E_b/N_o = 10$ dB
No	Wide	Wide	$1.2/T_b$	$1.2/T_b$	SQPSK	1.1 dB
Receive	Wide	Wide	$1.2/T_b$	$1.2/T_b$	SQPSK	1.1 dB
No	Wide	Wide	$1.2/T_b$	$1.2/T_b$	QPSK	0.7 dB
Receive	Wide	Wide	$1.2/T_b$	$1.2/T_b$	QPSK	0.6 dB
No	Wide	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	1.9 dB
Receive	Wide	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	2.1 dB
No	Wide	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	1.2 dB
Receive	Wide	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	1.2 dB
No	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	2.6 dB
Receive	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	SQPSK	2.8 dB
No	$1.4/T_b$	$1.4/T_b$	$1.4/T_b$	$1.4/T_b$	SQPSK	1.9 dB
Receive	$1.4/T_b$	$1.4/T_b$	$1.4/T_b$	$1.4/T_b$	SQPSK	2.0 dB
No	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	2.1 dB
Receive	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	$1.2/T_b$	QPSK	2.0 dB
No	$1.4/T_b$	$1.4/T_b$	$1.4/T_b$	$1.4/T_b$	QPSK	1.2 dB
Receive	$1.4/T_b$	$1.4/T_b$	$1.4/T_b$	$1.4/T_b$	QPSK	1.2 dB

5-pole Butterworth-amplitude characteristic, zero phase shift filters
Sampling detector
8-tap receive equalizer
T_b = Bit duration on inphase or quadrature
Tap spacing = T_b

SECTION IX
DUOBINARY SIGNALING TO IMPROVE BANDWIDTH UTILIZATION

This section studies the feasibility of improving bandwidth utilization by increasing the data rate of QPSK or SQPSK over a channel of restricted bandwidth. Applied to TDMA operation, this allows a higher burst data rate through the channel. Applied to FDMA operation, each transmitter and receiver can have a narrower bandwidth filter for a given data rate; hence, the frequency spacing between the FCMA signals can be reduced.

The increase in data rate is achieved as a consequence of applying the duobinary signaling concept[14]. The point of view adopted here is that of modifying the receiver processing without changing the QPSK or SQPSK modulation at the transmitter. Thus, an adaptive equalizer is utilized in the receiver to form the overall transfer function necessary for duobinary demodulation. As will be seen, this involves only minor changes from Section 5 with respect to the definition of error in the bit decision samples and the operation of the delay lock loop to optimize the sampling time.

Since improved bandwidth utilization is necessarily accompanied by an increase in required signal-to-noise ratio, the study concentrates on evaluating probability of error as a function of E_b/N_o. A method to reduce the required E_b/N_o by taking advantage of the inherent redundancy of duobinary is described. Finally, duobinary performance with rate-1/2 error correction coding is obtained; here, the objective is to reduce the E_b/N_o required to support a specified data rate, still achieving an improved bandwidth utilization compared with conventional coded QPSK.

9.1   DUOBINARY CONCEPT

We begin by describing the basic duobinary signaling concept, which is a modification of conventional binary signaling. The speed advantage of duobinary over conventional binary is due to the controlled utilization of intersymbol interference. Although only two levels are transmitted, three levels are received. The middle level at zero amplitude is interpreted as binary 1 (corresponding to a polarity transition in the transmitted signal) while the top and bottom levels are both interpreted as binary 0 (corresponding to no transition in the transmitted signal).

Thus, duobinary demodulation has a similarity to differential decoding, and the data is transmitted after differential encoding.

Figure 9-1 shows the idealized overall channel transfer function and the impulse response for duobinary signaling. Conventional binary demodulation of the received signal would sample at the peak, in which case the adjacent bits displaced in time by T_b would introduce significant intersymbol interference. Duobinary demodulation displaces the sampling time by $T_b/2$, and now there are three levels, +2, 0, -2, due to the controlled unit amplitude intersymbol interference from the preceding bit only.

776-3688
UNCLASSIFIED

(A)   (B)
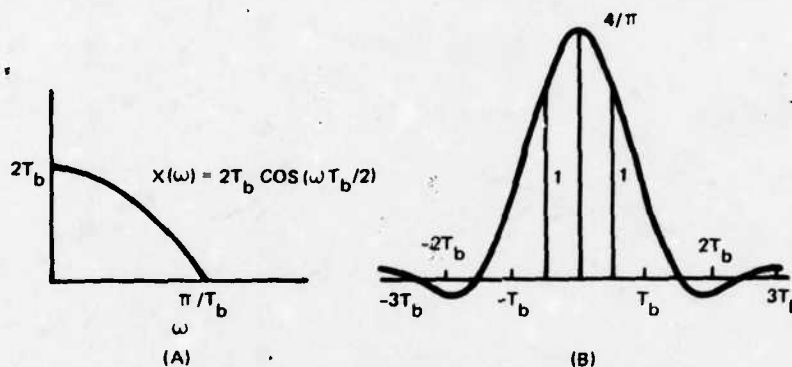
Figure 9-1.   (a) Duobinary Filter and (b) Its Impulse Response

Assuming white noise in the receiver, the E_b/N_o degradation is computed in reference 14 to be 2.1 dB when the idealized duobinary overall transfer function of Figure 9-1 is split equally between transmitter and receiver. However, in the application studied here, the transmitted spectrum is approximately flat out to $\omega = \pi/T_b$ due to the sharp cut transmit filter; hence, the spectrum must be shaped for duobinary entirely in the receiver. (The adaptive equalizer in the receiver performs this function.) Thus, for amplitude levels 2d, 0, -2d in the bit decision samples and noise density N_o, we have

$$\text{Transmitted power} = S = \frac{d^2}{2\pi T_b} \int_{-\pi/T_b}^{\pi/T_b} d\omega = d^2/T_b^2 \qquad (1)$$

$$\text{Noise power after receive filter} = \sigma^2 = \frac{N_o/2}{2\pi} \int_{-\pi/T_b}^{\pi/T_b} [2T_b \cos(\omega T_b/2)]^2 d\omega = T_b N_o$$

9-2

The distance to the decision threshold separating the levels is d; hence, the probability of error is characterized by the ratio

$$d/\sigma = \sqrt{ST_b/N_o} \qquad (3)$$

Since the probability of error for ideal binary transmission with conventional demodulation via a matched filter is characterized by the ratio $\sqrt{2ST_b/N_o}$, ideal duobinary spectrum shaping and demodulation entails 3 dB degradation in signal-to-noise ratio.

The duobinary concept is directly applicable to BPSK. It can also be applied to QPSK or SQPSK by combining two BPSK signals in phase quadrature, in the usual way. For QPSK, we assume independent differential encoding of the parallel I and Q bit streams, rather than mod-4 differential encoding, where the latter has the advantage of being invariant to a phase slip by 90°. The former is the only differential encoding applicable to SQPSK; however, the I and Q bit streams can be distinguished by their timing displacement of $T_b/2$.

Henceforth, only QPSK or SQPSK is discussed for duobinary signaling.

9.2   BASIC PERFORMANCE OF DUOBINARY

Duobinary performance in a restricted bandwidth channel can be evaluated by a computer simulation of the modulation/demodulation process. Here, performance is determined for the basic duobinary concept where the bit decision is independently made on the sampled amplitude for that bit; i.e., the inherent redundancy in the duobinary waveform is not exploited, as yet.

The computer simulation model described in Section 5.2.5 for QPSK in a bandlimited channel, with a bandlimited receiver, an adaptive equalizer, and a conventional sampling demodulator, is easily modified for duobinary demodulation. First, the operation of the delay lock loop for clock tracking is adjusted so that the sampling time to extract the bit decision amplitudes is defined to be coincident with the bit transitions, rather than displaced by $T_b/2$. Second, the decision-directed error for the LMS algorithm control of the equalizer is defined with respect to the three levels of duobinary, which are taken now to be -1, 0, +1. Thus, instead of a conventional binary decision on sample polarity, decision levels are placed symmetrically at ±0.5. The correct amplitude for a sample more positive than +0.5 is taken to be +1.0. (This is binary 0.) Similarly, the correct amplitude for a

sample more negative than -0.5 is taken to be -1.0. (This is also binary 0.)* The correct amplitude for samples in the range -0.5 to 0.5 is taken to be zero. (This is binary 1.)

After the tap weights of the adaptive equalizer have converged, the output noise power is obtained by a numerical integration of the receiver's power transfer function (receive filter in cascade with the equalizer) and the probability of error is averaged over the periodic data sequence, presuming Gaussian statistics and decision thresholds at ± 0.5, as described above. For binary 1, an error occurs if noise moves the amplitude outside the range from -0.5 to +0.5. Similarly, for binary 0, an error occurs if noise moves the amplitude into the range -0.5 to +0.5.

The simulation assumes 5-pole Butterworth filters, of various bandwidths. Results are presented in Table 9-I. The actual E_b/N_o is defined with respect to the power remaining after transmit filtering to the specified bandwidth. The degradation is the reduction in E_b/N_o to produce the same probability of error with ideal PSK. Since the duobinary demodulation presumes differential encoding, which is required in practice for conventional binary demodulation also, the degradation in Table I is taken with respect to ideal $\triangle$PSK.

Table 9-I. Duobinary Demodulation of QPSK With Decision-Directed Equalization in Receiver

Channel Condition	Transmit Bandpass	Receive Bandpass	Actual E_b/N_o	Degradation from Ideal $\triangle$PSK
No Distortion	$2/T_b$	$1/T_b$	12 dB	3.6 dB
No Distortion	$2/T_b$	$0.67/T_b$	13 dB	3.3 dB
No Distortion	$1/T_b$	$1/T_b$	12 dB	2.8 dB
No Distortion	$0.75/T_b$	$0.75/T_b$	12 dB	2.4 dB
No Distortion	$0.67/T_b$	$0.67/T_b$	12 dB	3.0 dB
No Distortion	$0.6/T_b$	$0.6/T_b$	14 dB	4.8 dB
$1 - j \sin(1.25\omega T_b)$	$0.67/T_b$	$0.67/T_b$	13 dB	4.0 dB
5-Pole Butterworth Filters 8 Tap Equalizer, Tap Spacing = T_b				

*As a refinement, the correct amplitudes of ± 1.0 can be made adaptive also.

It is seen that the performance is closely in accordance with idealized duobinary even for a bandwidth as narrow as $0.67/T_b$. Comparing with performance of conventional binary demodulation, which requires a bandwidth $1/T_b$, as seen in Section 5.2.5.2, it is found that duobinary demodulation enables approximately 50 percent greater data rate, at a modest penalty in E_b/N_o. A total data rate (summation of inphase and quadrature bit streams) equal to three times the bandpass is feasible, with approximately 3 dB degradation from ideal differentially encoded QPSK, in a nonsaturating channel.

9.3 MAXIMUM LIKELIHOOD DEMODULATION OF DUOBINARY

Section 9.2 has demonstrated feasibility of applying an adaptive equalizer in the receiver to form a duobinary channel response, thereby increasing the maximum data rate for QPSK signaling through a narrow bandpass channel. However, the required E_b/N_o is degraded by approximately 3 dB, as a consequence of the ternary decision for duobinary demodulation. That is, if v denotes the sample amplitude (soft decision), which ideally would have one of the levels −1, 0, 1 in the absence of noise, the duobinary decision is

$$.5 < v \implies \text{No transition; } + + \text{ sequence}$$

$$-.5 < v < .5 \implies \text{Transition } ; \left.\begin{array}{c} + \ - \\ - \ + \end{array}\right\} \text{sequences}$$

$$v < -.5 \implies \text{No transition; } - - \text{ sequence}$$

where the decision thresholds are midway between the ideal levels. With differential encoding, the + + and − − sequences of transmitted bits are both interpreted in the receiver as binary 0 for the data bit, while the + − and − + sequences are both interpreted as binary 1.

With duobinary demodulation, making a data bit decision on a single sample necessitates a ternary to binary mapping, and potentially usable information is inherently destroyed by this mapping. In the simplest utilization of the redundancy of duobinary, the ternary decision gives an error detection capability. For example, a decision that $.5 < v$ following a decision that $v < -.5$ in the previous sample is not consistent with the duobinary structure; hence, an error has been detected (however, its location is still unknown).

9.3.1 VITERBI ALGORITHM TO CORRECT ERRORS ON DUOBINARY CHANNEL

The redundancy inherent in the duobinary structure allows application of the Viterbi algorithm* as a mechanism for error correction, thereby decreasing the E_b/N_o degradation of duobinary demodulation.[15] It is assumed that the linear equalizer in the receiver forms an ideal duobinary response, such that the output due to a single pulse has two sample values equal to 0.5 with the remaining samples being zero. Thus, there is intersymbol interference from the preceding pulse only. Transmitting a sequence of pulses of amplitude ±1, a sample can take on one of the amplitudes .5 + .5 = 1, .5 - .5 = 0, or -.5 - .5 = -1.

The basis for the Viterbi algorithm is that the received amplitude sample depends only on the preceding and present pulse polarities. Then, the preceding pulse polarity specifies the present state of the channel, and for binary transmission, there are two possible states. The new state is specified by the polarity of the present pulse. Figure 9-2 gives the state transition diagram with associated correct received amplitude values (in the absence of noise).
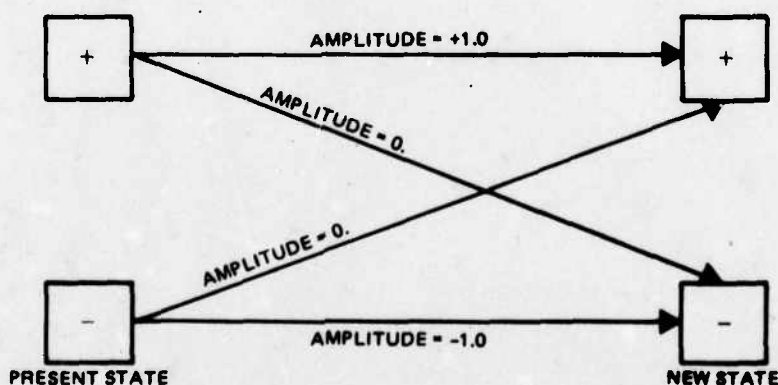
776-3589
(UNCLASSIFIED)

Figure 9-2. State Transition Diagram for Duobinary Channel

---

*Note, this does not mean a Viterbi decoder for a convolutional code. As used here, Viterbi algorithm implies a recursive computation exploiting a finite set of system states to trace the optimum path yielding the maximum likelihood sequence. Viterbi decoding is discussed for duobinary in Section 9.4.

Presuming additive Gaussian noise to each sample, the log likelihood of a received amplitude v is $-(v - a)^2/2$, where a is the correct amplitude (see Figure 9-2). If the noise is independent from sample to sample, the log likelihood is additive and forms a metric for the Viterbi algorithm, where the objective is to find the sequence of states with maximum metric. This sequence also minimizes the mean square error. However, only metric differences are needed; hence, we eliminate the common term $-v^2/2$. The resulting metric increment is given in Table 9-II, and shows how the soft decision v should be utilized by the Viterbi algorithm.

Table 9-II. Metric for Duobinary Channel

State Transition		Correct	Metric
Present	New	Amplitude	Increment
+	+	1.0	v-.5
+	-	0.	0.
-	+	0.	0.
-	-	-1.0	-v-.5

The Viterbi algorithm compares the metrics computed for the two possible paths leading to a new state, and only the higher metric is retained in storage for the new state. Also, this binary decision is then appended to the sequence of prior binary decisions on the retained path, and the new sequence is stored in the path memory for the new state. This comparison and updating process is performed for each possible new state. For the duobinary channel, there are just two states, and the Viterbi algorithm is quite simple. (The algorithm needs storage only for a single metric difference and a pair of path memories.) Figure 9-3 is a flowchart of the algorithm. (A similar algorithm is described in reference 16.)

The path memories introduce a decoding delay. The delayed sequence of decisions is differentially decoded to form the output binary data. Provided that the path memories are long enough, the two stored sequences will have remerged so that identical outputs are obtained from either memory. (Slightly better performance is achieved for a fixed decoding delay by taking the output from the path memory of the state with maximum metric, but this is a refinement causing extra complexity.)

ENTER

SOFT DECISION = V
METRIC DIFF = AM
PATH MEMORIES = I0, I1

A0 = V - .5
A1 = AM
A2 = AM - V - .5
J = SHIFT(I0) + 0
K = SHIFT(I1) + 1

A1 > A0 ?

NO

YES

A2 > 0 ?   NO

YES

A2 > 0 ?   NO

YES

AM = A2 - A1
I0 = K
I1 = K

AM = -A1
I0 = K
I1 = J

AM = A2 - A0
I0 = J
I1 = K

AM = -A0
I0 = J
I1 = J

EXIT

776-3590
(UNCLASSIFIED)
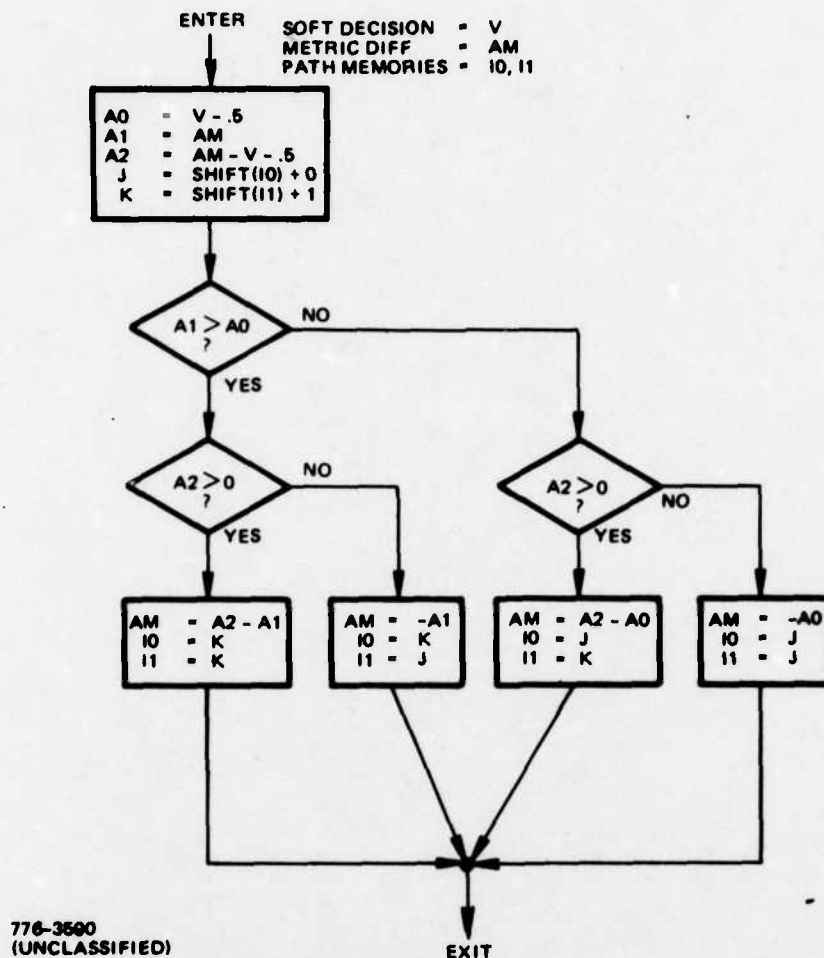
Figure 9-3. Flowchart of Viterbi Algorithm Correction for Duobinary
Demodulation

9.3.2   COMPUTER SIMULATION

The computer simulation model used in Section 9.2 for duobinary demod-
ulation was modified to add the Viterbi algorithm correction routine as depicted
in Figure 9-3.* The program causes the adaptive equalizer to form a duobinary
response with tap weight control from the decision-directed error of duobinary
demodulation at the E_b/N_o specified for equalization. This adaptive process is
unchanged by the addition of the Viterbi algorithm. After convergence, the tap weights

---

*The flowchart would be easier to follow if the two states each had a metric storage.
However, as mentioned, only the metric difference needs to be stored.

are frozen, and the bit decision samples from the equalizer output are input to the Viterbi algorithm routine. The inphase and quadrature bit decision samples are taken as parallel independent bit streams, with parallel Viterbi algorithm correction, so that storage is necessary for four states.

In this simulation, the error rate is measured by counting errors actually made by the Viterbi algorithm when noise is present at the E_b/N_o specified for data transmission. The output data is taken from path memory 10, which introduces a delay of 32 bits (the computer word length). As mentioned previously, the output from the path memory is differentially decoded to give the output data.

Simulation results are plotted in Figure 9-4, where E_b/N_o is defined with respect to average signal power at the output of the transmit filter. The abscissa is the E_b/N_o specified for data transmission. The E_b/N_o specified for equalization is given for each curve. In the linear channel without saturation, the measured error rate is seen to be very close to ideal differentially decoded PSK. Thus, the Viterbi algorithm almost completely recovers the 3 dB degradation for basic duobinary demodulation of BPSK or QPSK. The total data rate with QPSK (sum of inphase and quadrature) is 3 bps/Hz for the channel bandwidth of Figure 9-4.

9.3.3   SATURATION IN CHANNEL

If a hard limiter exists in the narrow bandpass channel, duobinary demodulation is severely degraded. Since the transmit filter prior to the hard limiting does not pass the data sidebands at $\pm 0.5/T_b$, the filter output due to alternating bits tends to have an amplitude close to zero. This is proper for duobinary, and there is no problem in a linear channel. However, the hard limiting forces the transmitted signal to a constant envelope. Since the bit stream switches polarity half the time with random data, a heuristic argument based on wasted energy suggests a minimum additional degradation of 3 dB is expected due to hard limiting.

Simulation results for duobinary demodulation with Viterbi algorithm correction are presented in Figure 9-4 presuming a hard limiting channel. The measured error rate results show roughly 3 dB displacement from the results for a linear channel, in accordance with the heuristic argument given above. However, without the Viterbi algorithm correction, the degradation due to hard limiting is found to be considerably more than 3 dB.
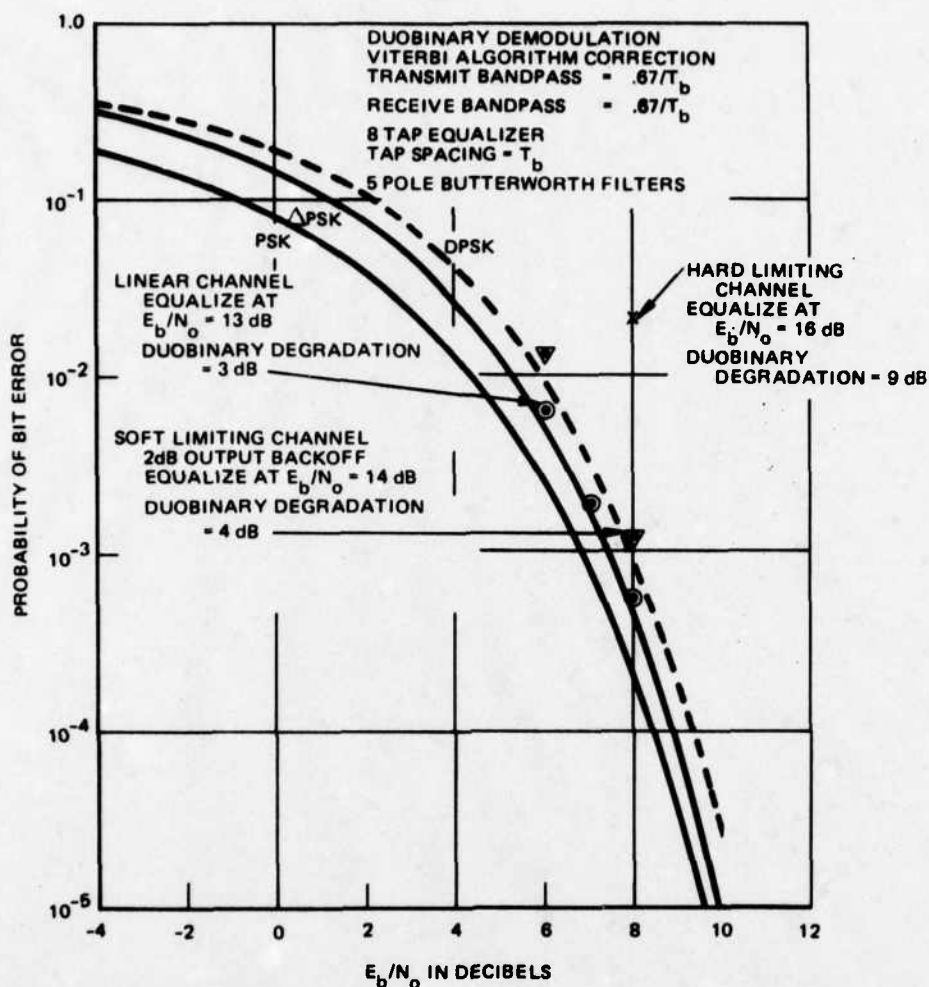
Figure 9-4. Simulation Results for QPSK Duobinary

Next, we assume soft limiting in the channel. Now, when the signal amplitude is high, the soft limiter is driven into saturation. However, when the signal amplitude is low after a polarity switch, the limiter is backed off into the linear region. An error function limiter characteristic is assumed, so that

$$v_{out} = erf(A \, v_{in}) \qquad (4)$$

where v_{in} is the input envelope (magnitude of complex sample), v_{out} is the output envelope, and A adjusts the backoff (A → ∞ gives hard limiting, while A → 0 gives linear operation). There is no AM to PM conversion in this limiter model, so that the phase of the limiter output is identical to that of the limiter input. Results of a computer simulation are given in Figure 9-4. Choosing the gain into the limiter to

9-10

produce an output average power backed off by 2 dB, performance is close to that
achieved over a linear channel.

Figure 9-5 gives the output average power as a function of the input gain to
the soft limiter. The output backoff from saturation can be viewed as an effective
degradation in the sense that the peak available channel power is not exploited.
Including the degradation as found from Figure 9-4, an effective total degradation can
be defined as the sum (in decibels) of the output backoff and the performance
degradation from ideal differentially encoded QPSK performance. The estimated
effective total degradation is plotted in Figure 9-5, and shows a near constant value
over a wide range of input gain.

Some runs were made assuming SQPSK instead of QPSK. Very similar
results were obtained.
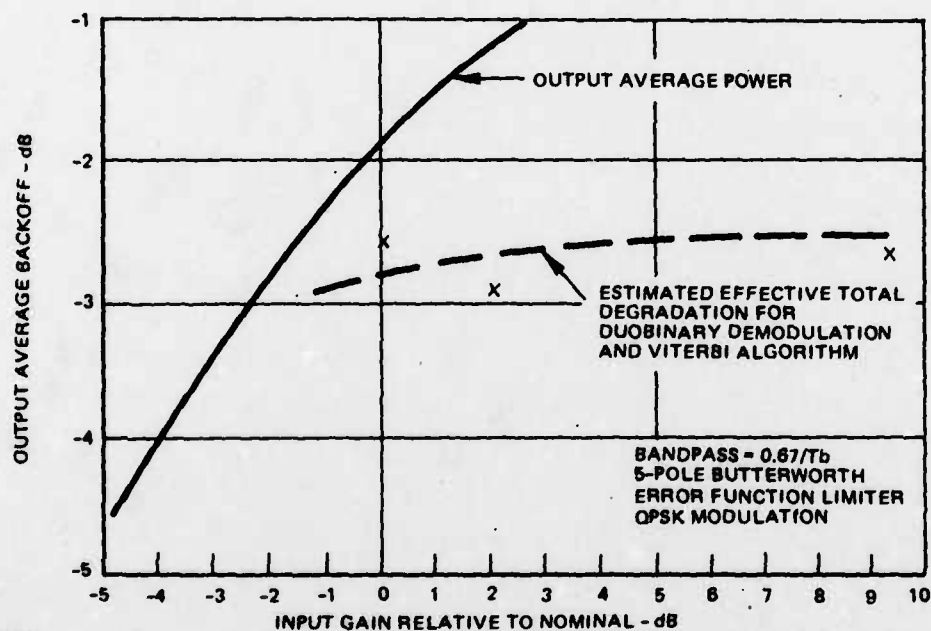
Figure 9-5. Estimation of QPSK Duobinary Performance in Narrowband
Channel With Soft Limiting

9.4   RATE-1/2 CODING GAIN ON DUOBINARY CHANNEL

The question is now raised whether error correction coding can be utilized with duobinary to achieve a coding gain (i.e., reduction in E_b/N_o) comparable to that attainable with conventional BPSK or QPSK. To investigate this question, we first assume an ideal duobinary channel in the following and develop a rate-1/2 convolutional code/Viterbi decoder simulation which is optimized for duobinary demodulation. The Viterbi decoder for duobinary has a different metric and an extended number of internal states, compared with a standard Viterbi decoder designed for the memoryless PSK channel (see Section 4).

Then, using the computer simulation described in Section 9.2, a narrow-band channel transmitting staggered quadriphase (SQPSK) is equalized for duobinary demodulation, and the rate-1/2 convolutional code/Viterbi decoder simulation developed for the ideal duobinary channel is employed. Saturation is allowed in the channel after the transmit filter. The backoff from hard limiting is adjustable.

9.4.1   IDEAL DUOBINARY CHANNEL AND ASSOCIATED METRIC

The ideal duobinary channel is defined as producing intersymbol interference from the present transmitted bit only. On any sample, this intersymbol interference has an amplitude equal to the amplitude contributed from the new bit. Therefore, with binary modulation, a noise-free sample can take on one of the values -1, 0, 1, as illustrated in Figure 9-2. The associated metric for duobinary is given in Table 9-III, which applies independently to the inphase and quadrature components for QPSK. (Table 9-III is the same as Table 9-II , except for representing the transmitted bits as binary 0, 1 instead of as positive or negative polarity.)

Table 9-III. Metric for Duobinary Channel With Coding

Present Transmitted Bit	New Transmitted Bit	Correct Sample Value	Metric Increment
0	0	1.0	v - .5
0	1	0.	0.
1	0	0.	0.
1	1	-1.0	-v - .5

For comparison, ideal PSK has no intersymbol interference, and the metric increment is $\pm v$, depending on the hypothesized correct sample polarity.

9.4.2   STATES FOR VITERBI DECODER ON DUOBINARY CHANNEL

On a memoryless channel, such as ideal PSK, a rate-1/2 Viterbi decoder for a binary convolutional code of constraint length K has 2^{K-1} states. This is true because the metric increment is determined from the new coded bit, which is a function of the last K - 1 data bits and the hypothesized new data bit. The state is defined by the last K - 1 data bits.

With duobinary, which introduces memory into the channel, the metric increment is determined from both the present coded bit and the new coded bit. The present coded bit is a function of the last K data bits, while the new coded bit still is a function of the last K - 1 data bits and the hypothesized new data bit. Consequently, the Viterbi decoder for a code of constraint length K must have 2^K states when applied to the duobinary channel. The number of states is doubled to handle the intersymbol interference of duobinary. With the aid of Table 9-III, the metric increment is determined for each possible state transition in the decoder. The Viterbi algorithm finds the sequence of states yielding the maximum metric.

To illustrate the Viterbi decoder for duobinary, let K = 5 so that the present coded bit is a mod-2 sum of selected taps over the last 5 data bits*. Figure 9-6 shown an example of the state transitions, the present and new coded bits, and the resulting sample amplitude for duobinary. The metric increment for each state transition is found from Table 9-III.

Cataloguing all possible state transitions and the corresponding metric increments, the Viterbi decoder for duobinary can be simulated. As suggested by Figure 9-6, for a rate-1/2 code, the present states are grouped in pairs differing in the oldest data bit, and the new states are grouped in pairs differing in the newest data bit.

The present study is specialized to a rate-1/2 convolutional code, transmitting the respective coded bits from the pair of independent mod-2 sums on independent inphase and quadrature components of QPSK with duobinary demodulation. In this model, there is no intersymbol interference between the independent coded

_____

*Note for simplicity, we illustrate just one set of taps producing the coded bits. There are two independent sets for a rate-1/2 code.

PRESENT CODED BIT

NEW CODED BIT

MOD-2

MOD 2

1 0 1 1 0

0 1 1 0 0

OLDEST DATA BIT

PRESENT DATA BIT

00 = +1.0

10 = 0.

NEWEST DATA BIT

0 0 1 1 0

NEW STATE

PRESENT STATE

RESULTING DUOBINARY AMPLITUDE

1 0 1 1 0

01 = 0.

11 = -1.0

0 1 1 0 1

776-3593
UNCLASSIFIED

0 0 1 1 0
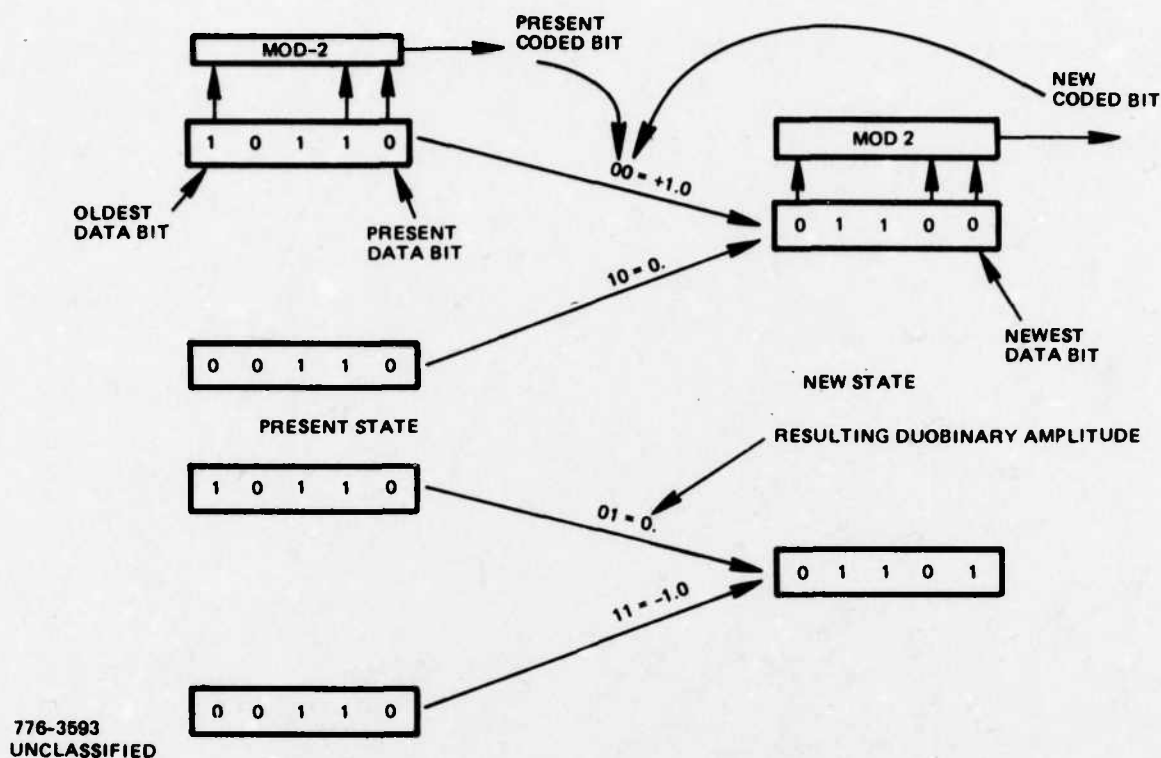
Figure 9-6. Illustration of State Transitions, Corresponding Present and New
Coded Bits, and Resulting Duobinary Amplitude

bit streams. Duobinary creates intersymbol interference only between the present
coded bit and the new coded bit on each independent stream.

9.4.3   COMPUTER SIMULATION MODEL FOR IDEAL DUOBINARY CHANNEL

A simulation was carried out assuming an ideal QPSK channel with ideal
duobinary response. The sampling rate is equal to the coded bit rate. Table 9-III
defines the sample amplitude resulting from the present and new coded bits trans-
mitted over the channel. With a rate-1/2 error correcting code, the coded bits from
the pair of independent mod-2 sums are transmitted, respectively, as the inphase
and quadrature components. Intersymbol interference does not occur between inphase
and quadrature.

To begin, there is no error correcting code, and the Viterbi algorithm
has only two states (i.e., K = 1). Simulation results for transmitting random data
are plotted in Figure 9-7 and closely approach the ideal differentially encoded PSK

1.0

TRANSMISSION OF
DIFFERENTIALLY
ENCODED DATA

10^{-1}

△PSK

DPSK

PSK

NO ERROR
CORRECTION
VITERBI
ALGORITHM
PATH MEMORY
= 32 BITS

10^{-2}

PROBABILITY OF BIT ERROR

CONVOLUTIONAL
RATE - 1/2 CODE

VITERBI ALGORITHM
CONSTRAINT
LENGTH = 5
PATH MEMORY = 25 BITS

$\left\{\begin{array}{c}10011\\11101\end{array}\right\}$ TAPS

10^{-3}

10^{-4}

10^{-5}

-4   -2   0   2   4   6   8   10   12

E_b/N_o IN DECIBELS
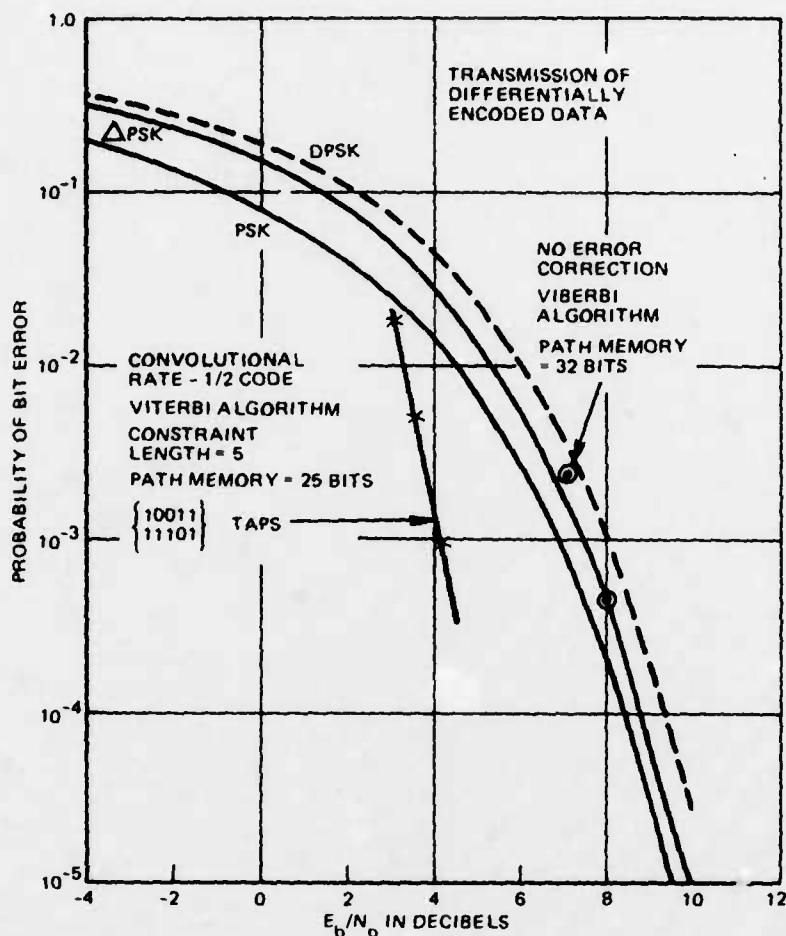
776-3594
UNCLASSIFIED

Figure 9-7. Ideal Duobinary

curve*. These results are similar to Figure 9-4, for a simulation of a QPSK narrow-
band channel adaptively equalized to approximate the duobinary response. The
error rate is measured by counting erroneous data bits output from the Viterbi
algorithm at the specified E_b/N_o. Note, differential encoding of the data bits is
presumed**.

---

*E_b/N_o is defined with respect to average power at the receiver. For amplitudes
 -1, 0, 1 of duobinary, the average power is 0.5.

**Differentially encoded data and a transparent convolutional code is an easy way to
 accommodate 180° phase ambiguity of BPSK. QPSK has 90° ambiguity also, and
 this must be resolved by distinguishing the inphase and the quadrature components.

9-15

Next, a rate-1/2 convolutional code of constraint length 5 is employed. The selected code is the same as found optimum for the memoryless PSK channel. (Section 9.5 shows this to be a good code for the duobinary channel also.) The code is defined by the two sets of taps for the mod-2 sums, as listed in Figure 9-7. Simulation results for error rate with differentially encoded data are plotted in Figure 9-7. These results are observed to be only slightly worse (by roughly 0.5 dB) than previously found for the same convolutional code and a Viterbi decoder on the memoryless PSK channel. (Compare with Figure 4-1, which, by the way, does not assume differentially encoded data.)

9.4.4 COMPUTER SIMULATION OF SQPSK CHANNEL EQUALIZED FOR DUOBINARY

The computer simulation model of Section 9.2 for SQPSK in a channel adaptively equalized for duobinary demodulation was modified to include the rate-1/2 convolutional code/Viterbi decoder. Because of anticipated difficulties in duobinary decision-directed equalization at low E_b/N_o, the adaptive equalization control was derived from correct bits. (The error for the LMS algorithm is defined with respect to the correct sample amplitude given in Table 9-III as a function of the present and the new transmitted bits.)

The transmit and receive filters are Butterworth, and saturation in the channel is allowed, as defined by (4). The simulation proceeds by first equalizing to form the least mean square approximation to the duobinary response. After convergence, the equalizer tap weights and the bit sampling time are frozen, and the Viterbi decoder is activated. The noise consists of independent Gaussian samples added to the transmitted signal samples prior to the receive filter. The transmitted signal is periodic, with 32 data bits in one period and 32 samples per data bit. There are 32 coded bits on both the inphase and the quadrature components for a rate-1/2 error correction code. The noise is treated as periodic in the receive filter; however, new noise samples are generated for each successive period. The error rate is obtained by counting errors in the decoder's output. The data is differentially encoded prior to being rate-1/2 convolutionally encoded for error correction.

Simulation results are plotted in Figure 9-8 for both linear operation, soft limiting (backoff equal to 2 dB), and hard limiting. Comparing with Figure 9-7, the performance for linear operation is roughly 0.5 dB degraded from ideal duobinary. Under hard limiting, there is a further degradation of approximately 2 dB. At a backoff of 2 dB, there is approximately 0.5 dB degradation compared with linear operation.

Comparison should also be made between Figure 9-8 here and Figure 9-4. The latter gives performance for uncoded data but using a simple Viterbi algorithm to exploit the duobinary structure. At 10^{-3} error rate, the error correction code with constraint length 5 gives approximately 3 dB coding gain for linear operation, and somewhat more for hard limiting operation. This coding gain is essentially identical to that realized at 10^{-3} error rate for the same error correction code used over the ideal PSK channel.

9.4.5 SOFT DECISION QUANTIZING FOR DUOBINARY VITERBI DECODING

To implement a Viterbi decoder, the soft decisions must be quantized. Examples of 3-bit quantization (8 levels) for PSK demodulation, with uniform spacing of levels, are given in Section 3.9. The quantizer spacing is not critical. One example sets the spacing equal to the distance between the positive and the negative signal amplitudes (in the absence of noise) divided by the number of levels (equal to 8).

With duobinary demodulation, the distance between zero amplitude and either the positive or the negative signal amplitude corresponds to the distance between positive and negative amplitudes for PSK demodulation. Thus, it is reasonable to quantize to 4 bits (16 levels) with the quantizer spacing still set equal to the distance between the positive amplitude and the negative amplitudes divided by the number of levels (equal to 16). If the signal amplitude is unity, this yields a quantizer spacing of 0.125. Figure 9-9 plots the quantizer characteristic.

Note that if the quantizing levels are assigned a sign/magnitude binary code, the most significant magnitude bit gives a hard duobinary decision (i.e., 0 denotes magnitude less than 0.5, and 1 denotes magnitude greater than 0.5). The sign bit gives no information for hard duobinary decisions unless the redundancy in the duobinary structure is exploited for error detection.
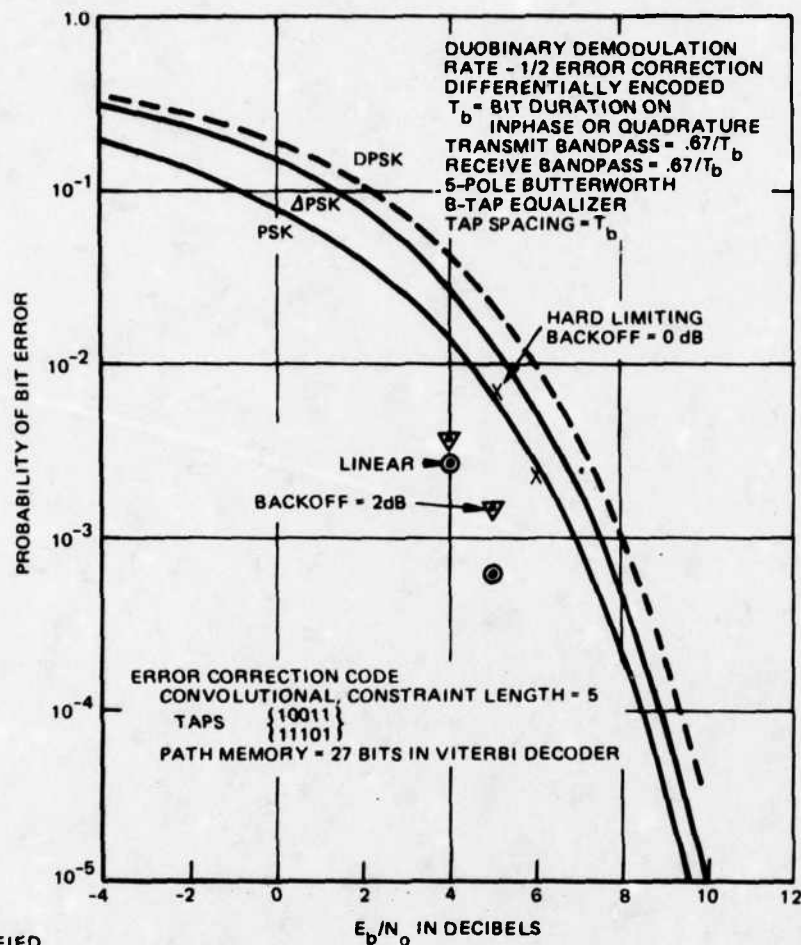
Figure 9-8. Simulation Results for SQPSK Duobinary

The computer simulation was modified to quantize the duobinary soft decisions according to Figure 9-9, for both LMS algorithm control of the equalizer and Viterbi decoding. For the linear case, the resulting error rate at $E_b/N_o = 5$ dB was about the same as shown in Figure 9-8. (Actually, the error rate with quantized soft decisions turned out to be somewhat lower, but the difference is not statistically significant.)
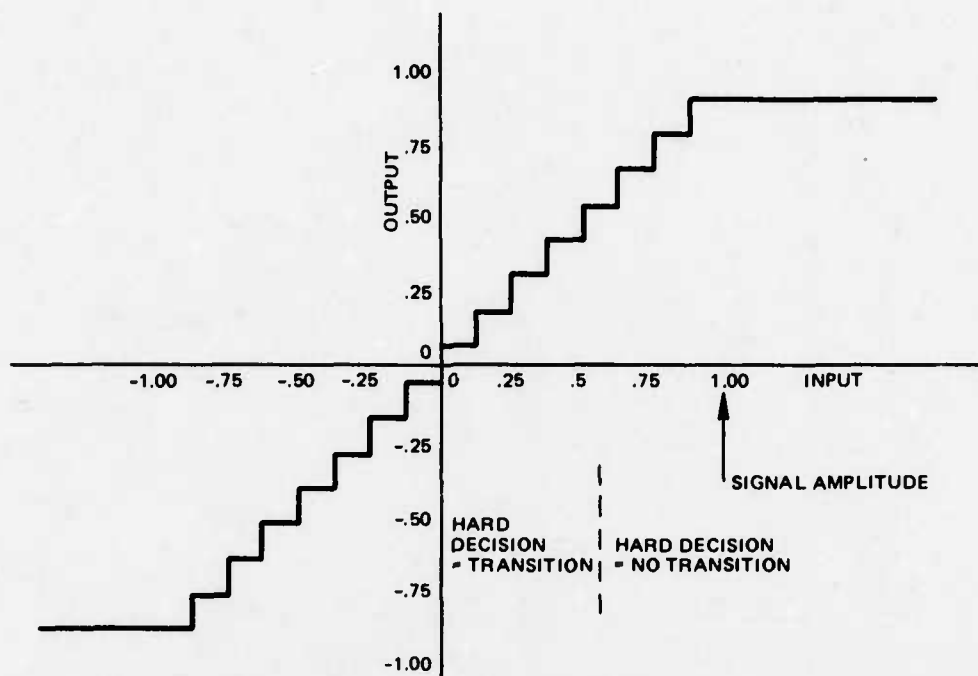
Figure 9-9. Soft Decision Quantizing for Duobinary

9.5   OPTIMUM RATE-1/2 CONVOLUTIONAL CODE FOR QPSK DUOBINARY

Section 9.4 demonstrated that a rate-1/2 convolutional code optimized for the memoryless binary channel also works well when applied to the QPSK duobinary channel. The question remains whether improved performance on the duobinary channel could be obtained from a different code. To help answer this question, the most likely error in decoding on the duobinary channel needs to be identified; then, the prospects for improvement can be assessed.

This investigation assumes an ideal duobinary channel response (Section 9.4.3) where the intersymbol interference causes the present transmitted bit (code symbol) to be linearly added to the new bit. With QPSK, the ideal duobinary response applies independently to the inphase and the quadrature (I and Q) bit sequences, and there is no interaction between the two. The channel output is ternary: +1 results from the ++ sequence; 0 from +- or -+; and -1 from --. Independent white Gaussian noise is added to the channel output.

9-19

A rate-1/2 convolutional code generates a pair of symbols as a function of each new data bit and the encoder state specified by the $K-1$ previous data bits, where K denotes the constraint length. For QPSK, one symbol if transmitted on I, and the second symbol is transmitted on Q. Convolutional coding on the ideal QPSK duobinary channel is modelled as shown in Figure 9-10. The delay of one bit due to the inter-symbol interference of ideal duobinary causes the present symbol to be added (algebraically, not mod-2) to the new symbol, in parallel on I and Q.

776-3579
UNCLASSIFIED
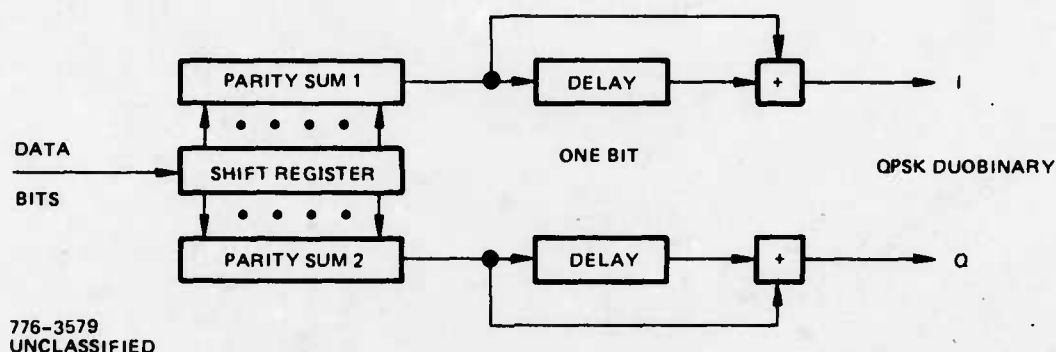
Figure 9-10. Model of Ideal QPSK Duobinary Channel With Rate-1/2 Coding

Figure 9-11 shows an example of a random bit sequence and its conversion to a ternary waveform as a consequence of the duobinary response. The duobinary has zero amplitude when the bit sequence has a transition, and positive or negative amplitude when the binary stream does not switch. Figure 9-11 has been drawn with the duobinary timing offset to emphasize that the binary channel is converted to duobinary when the sampling is displaced by half the bit duration.
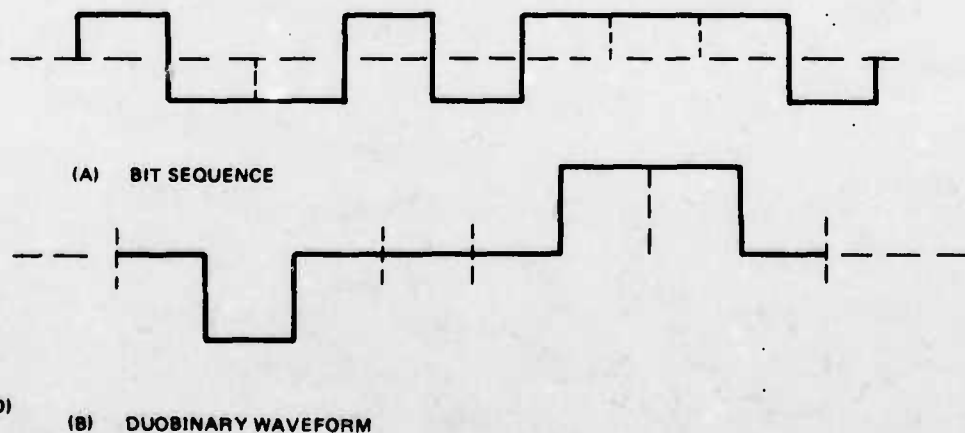
(A)   BIT SEQUENCE

776-3287
(UNCLASSIFIED)

(B)   DUOBINARY WAVEFORM

Figure 9-11. Bit Sequence and Corresponding Duobinary

9.5.1   ASYMPTOTIC PROBABILITY OF ERROR FOR DUOBINARY

As a lead in to the discussion of error correction coding for duobinary, we examine the probability of error in distinguishing two different bit sequences for the coherent duobinary channel. The energy in the difference between the waveforms is the significant parameter when optimum demodulation is performed assuming additive white Gaussian noise in the receiver. Figure 9-12 shows an example of two bit sequences differing by one bit, and the resulting difference between the duobinary waveforms.

If the binary waveform has unit power, the amplitude levels are ± 1. The difference between the two bit sequences has amplitude 2 or -2 for the single bit duration where the bit is changed; hence, the energy in the difference waveform is 4. If the duobinary waveform has unit power, the ternary amplitude levels are $\pm \sqrt{2}$ and 0. The difference between the two duobinary waveforms has amplitude $\sqrt{2}$ for two bit durations; hence, the energy is 4 also.

With a maximum-likelihood demodulation process, duobinary asymptotically has the same error probability as binary, at the same average power and assuming white noise in the receiver. The Viterbi algorithm is an implementation of maximum-likelihood demodulation. Actually duobinary has a somewhat higher error rate than binary because an incorrect waveform can have more than one bit changed, yet the energy in the difference is still 4.

Suboptimum demodulation of duobinary utilizes only the receiver waveform over a single bit duration, and there is an asymptotic penalty of 3 dB, since the distance between adjacent amplitude levels is $\sqrt{2}$ for duobinary compared with 2 for conventional binary.

The error mechanism can be characterized even more directly by making use of a property of duobinary, that it is a form of differential encoding of the bit sequence. That is, a transition in the bit sequence produces zero amplitude, while no transition gives full amplitude (positive or negative depending on the prior bits). Suboptimum demodulation of duobinary which maps the ternary decision for each bit into a binary output is based on this observation. One way to ensure a discernible difference between the duobinary waveforms corresponding to different bit sequences is now evident; one sequence should be made to have transitions where the second does not, and vice versa. Of course, for some pairs of sequences, one duobinary waveform can be positive where the second is negative, and this yields an even greater waveform difference, if this property can be exploited.

+1.0

-1.0

(A)   BIT SEQUENCE

(B)   BIT SEQUENCE (ONE BIT IS CHANGED)

$+\sqrt{2}$

0

$-\sqrt{2}$

(C)   DUOBINARY

(D)   DUOBINARY (ONE BIT IS CHANGED)

776-3288   (E)   DIFFERENCE BETWEEN DUOBINARY WAVEFORMS
(UNCLASSIFIED)
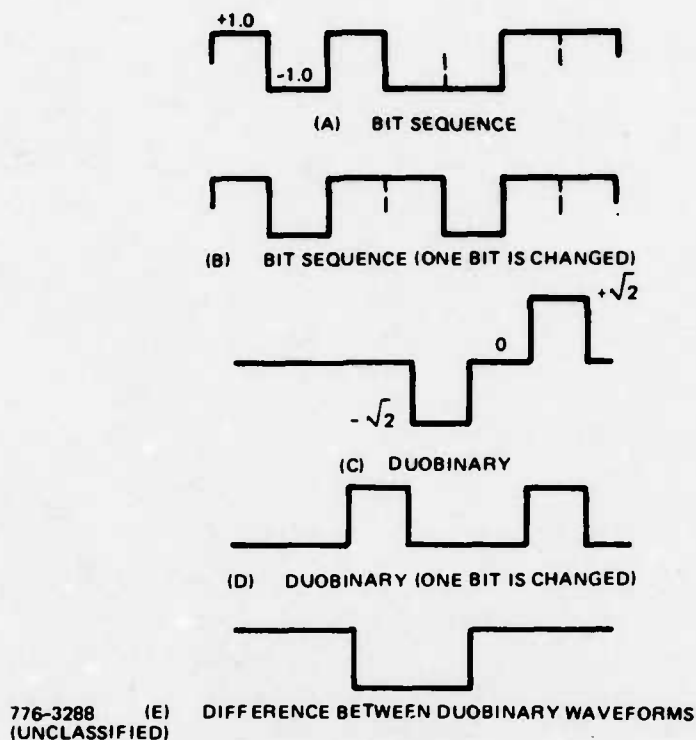
Figure 9-12. Mechanism of Error With Duobinary

An artificial example can be given where the 3 dB improvement from maximum-likelihood demodulation of duobinary is not realized. Postulate an alternating bit sequence, which yields a zero amplitude duobinary waveform. If the alternating bit sequence is complemented after some point, the duobinary waveform has a single nonzero amplitude, and the energy in the difference waveform is half the normal value. Figure 9-13 illustrates this artificial case. Note that when the alternations in the bit sequence cease, the missing energy in the difference waveform finally occurs. Therefore, the maximum-likelihood demodulation process for duobinary requires sufficient path memory storage (i.e., decoding delay) that the probability of having the artificial case is sufficiently low. With random data, the probability of getting 20 alternating bits is 10^{-5}; hence, the delay should certainly exceed 20 bits for the typical probabilities of error of interest.

(A)   BIT SEQUENCE (ALTERNATING BITS)

(B)   BIT SEQUENCE COMPLEMENTED AT BIT NO. 4

(C)   DUOBINARY CORRESPONDING TO (A)

776-3289   (D)   DUOBINARY CORRESPONDING TO (B)
(UNCLASSIFIED)
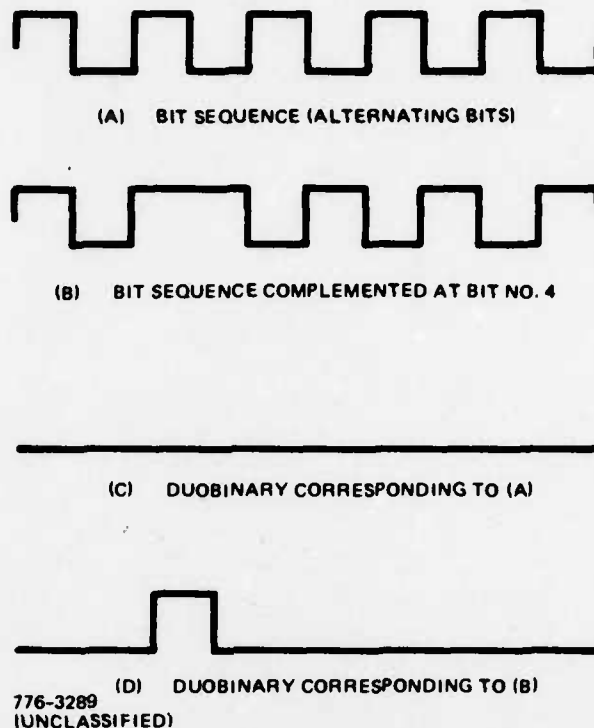
Figure 9-13.   Artificial Case Where Improvement From Maximum Likelihood
Demodulation of Duobinary is not Realized

9.5.2   OPTIMIZATION OF RATE-1/2 CONVOLUTIONAL CODE FOR DUOBINARY

Applying the above principle for ensuring a difference between duobinary
waveforms, we can find the optimum rate-1/2 convolutional code for QPSK duobinary
operation. Comparing any pair of code words to evaluate the probability of error,
we wish to maximize the total number of transitions which occur in one word but not
in the other. (Note, a word consists of two parallel binary streams as shown in
Figure 9-10.)

Now, observe that the transitions in a code word correspond to 1's in
the differentially decoded code word. Since a unit shift of any code word still is
a code word and differential decoding is mod-2 addition of the word with a unit
delay of itself, the differentially decoded code word must be itself a code word.

Requiring that one code word have as many 1's as possible where the second code word does not, and vice versa, is equivalent to requiring that the mod-2 sum of the two code words have as many as 1's as possible. But, again, this mod-2 sum yields a code word. Thus, the final conclusion is that the optimum rate-1/2 convolutional code for QPSK duobinary should maximize the minimum weight of any code word; i.e., should maximize the free distance.

The, perhaps surprising, result is that the same rate-1/2 convolutional code which is optimized for the memoryless channel by maximizing the free distance is also optimum for the QPSK duobinary channel in the sense of maximizing the number of non-simultaneous transitions.

A transparent code contains the all 1's code word, hence, a word with continual transitions. This code word produces zero amplitude after the duobinary response. Comparing this code word with any other, the minimum energy (in appropriate units) in the waveform difference is obviously equal to the free distance of the code. However, it is possible that the minimum energy in the waveform difference may exceed the free distance for a non-transparent code, since the zero-amplitude waveform cannot occur. Then, over the duobinary channel, a non-transparent code would be superior to a transparent code with the same free distance. (Of course, synchronization is more difficult with the non-transparent code because a 180° phase shift in the receiver cannot be ignored.)

9.5.3   COMPUTATION OF MINIMUM METRIC DIFFERENCE

The rate-1/2 Viterbi decoder for the duobinary channel can be utilized to find the minimum metric difference, which by the argument above is proportional to the energy in the difference between the correct waveform and the most closely matching incorrect waveform. The metric increments used in the decoder are given in Table 9-III, and the constant of proportionality is 0.5 with signal amplitudes of -1, 0, 1. (This corresponds to an average power of 0.5.)

For a constraint length 5 code, which is non-transparent and has free distance equal to 7, the minimum metric difference, adjusted to assume unit power in the signal for a random data bit sequence, is found to be 8, whereas the theory above claims a lower bound of 7. (This is the free distance multiplied by the constant of proportionality.)

For a constraint length 7 code, which is transparent and has a free distance equal to 10, the minimum metric difference is found to be 10.

Thus, the conjecture that a non-transparent code should be better than a transparent code of the same free distance on the duobinary channel appears to be true.

9.5.4 ASYMPTOTIC VERSUS ACTUAL CODING GAIN

The code word comparison with the minimum metric difference dominates probability of error asymptotically as E_b/N_o grows large. For equal powers in a duobinary signal and in a conventional binary signal, and using the same rate-1/2 convolutional code, the minimum metric difference for conventional binary is twice that for duobinary. Thus, as E_b/N_o becomes large, duobinary is asymptotically inferior by 3 dB.

The minimum metric difference occurs infrequently, however, with a random data bit sequence over the duobinary channel, and most metric differences are higher; possibly even exceeding that for conventional binary. At a probability of error typically of interest, say 10^{-5}, the E_b/N_o difference with coding between duobinary and conventional binary should be considerably less than 3 dB. This is borne out by simulation results in Figures 9-7 and 9-8.

9.5.5 CONCLUSIONS

A basis for optimizing a rate-1/2 convolutional code for the QPSK duobinary channel has been described. To optimize the worst case leading to the most likely decoding error, the code should have maximal free distance, the same criterion of optimality for the conventional binary channel. However, a non-transparent code is somewhat better than a transparent code of the same free distance.

Asymptotically, the worst case for a decoding error indicates a 3 dB degradation for duobinary, compared with conventional binary. However, the worst case occurs infrequently with random data, and the actual degradation at typical error rates of interest is small, according to simulation results.

9.6   RATE-3/4 VITERBI DECODING ON DUOBINARY CHANNEL

Section 4.4 describes a technique for converting a rate-1/2 convolutional code into a rate-3/4 code. One reason to consider a rate-3/4 code is for application to duobinary transmission with the objective of increasing the data throughput. As shown in Section 9.3, QPSK duobinary can transmit 3 bps/Hz uncoded, and with a rate-3/4 code, the data rate is still 2.25 bps/Hz. This exceeds the throughput of conventional QPSK even without coding.

The Viterbi decoder for duobinary must take into account the channel memory, which extends to the previous symbol (separately on I and Q for QPSK). With a rate-1/2 convolutional code, the channel memory of duobinary doubles the number of decoder states, since the previous symbol is determined from one additional old data bit (see Figure 9-6).

When the rate-1/2 code is converted into a rate-3/4 code by erasures, as described in Section 4.4, the Viterbi decoder for the duobinary channel is more complicated. Because of the erasures, two additional old data bits are sometimes required to determine the previous symbol; this multiplies the number of states by four. Furthermore, the metric increment is different for each of the 3 data bits corresponding to a block of 4 symbols.

9.7   CONCLUSIONS

A significant performance improvement for duobinary signaling in a narrow bandpass is derived from the Viterbi algorithm utilized to exploit the redundancy inherent in the duobinary structure. The receiver concept is to have an adaptive linear equalizer which forms the duobinary response, with tap weights controlled from the decision-directed error of duobinary demodulation. The soft decisions from the duobinary demodulator (which has a ternary output) are then input to a rather simple two-state Viterbi algorithm, which assumes the metric for an ideal duobinary channel.

In a linear channel without saturation, performance almost identical with ideal differentially encoded PSK is observed in a bandpass equal to $0.67/T_b$. This means 3 bps/Hz with QPSK or SQPSK. Thus, the Viterbi algorithm almost completely removes the duobinary degradation. These results are obtained by simulating the decision-directed LMS algorithm for equalizer adaptation and the Viterbi algorithm for error correction. The equalizer will remove phase distortion in the channel as well as forming the duobinary response.

In a saturating channel, the duobinary performance is degraded. With hard limiting, the degradation using the Viterbi algorithm is about 3 dB, which is heuristically explained as energy waste. With soft limiting (assumed as an error function), where the channel is driven into saturation when the signal has maximum amplitude, performance is close to that observed on the linear channel. Of course, the output average power backoff still has to be interpreted as an effective degradation, in the sense that the peak available power is not being exploited. The computer simulation results for QPSK and SQPSK suggest that the effective total degradation will be approximately 3 dB over a considerable range of input level to the soft limiter. This still gives an efficient technique for achieving a high QPSK/TDMA or SQPSK/TDMA burst rate in a narrow bandwidth (3 bps/Hz), without imposing a requirement for stringent power control on the ground transmitter.

With a rate-1/2 convolutional code of constraint length K, the Viterbi decoder for duobinary has 2^K states (rather than the 2^{K-1} states needed on the memoryless channel); hence, it is completely different from the standard decoder. Designing the Viterbi decoder with the metric appropriate for duobinary, the coding gain (reduction in required E_b/N_o) is close to that realized on the memoryless PSK channel. This means a 50 percent increase in data rate for duobinary also when error correction coding is employed. The bandwidth utilization is 1.5 bps/Hz for QPSK or SQPSK, for a rate-1/2 error correction code. Improved bandwidth utilization, 2.25 bps/Hz, would be obtained for a rate-3/4 code; however, the Viterbi decoder is more complicated.

Even though duobinary implies amplitude modulation, it is feasible to limit the transmitted SQPSK signal after narrowband filtering. The performance with coding is degraded by roughly 2 dB as a consequence of hard limiting, referenced to E_b/N_o at the receiver. The degradation is explained as an energy waste.

There are many questions which remain to be studied to assess the practical feasibility of duobinary with error correction coding. For instance,

1. Is there a benefit from nonuniform spacing when quantizing the duobinary soft decisions?

2. Will adaptive equalization be successful when controlled from the modem decisions at low E_b/N_o, or will the more reliable Viterbi decoder output be needed?

3. Is a training sequence needed for start up of the adaptive equalization?

4. Can a fixed filter of proper response be utilized?

The preliminary conclusion is that the duobinary concept is very attractive for improved bandwidth utilization.

It is interesting to compare the duobinary technique with the use of 8-phase signaling, which theoretically also increases the data rate by the factor 1.5. However, the theoretically required E_b/N_o for 8-phase signaling is approximately 4 dB higher than for QPSK, and intersymbol interference is much more deleterious.

SECTION X
CONCLUSIONS AND RECOMMENDATIONS

10.1   CONCLUSIONS

This section summarizes the major conclusions which have been noted during the study, based on extensive analyses and computer simulations.

• The phase distortion characteristics of the DSCS terminals and satellites will not significantly degrade performance of the 10 Mbps QPSK modems.

• The bandwidth of the 70 MHz IF of the COMTECH converters is wide enough (approximately 50 MHz) to support QPSK signaling at a data rate of 80 Mbps (uncoded). The 700 MHz wideband IF is not required for operation with the Phase III satellite channels.

• Error correction coding (which reduces the data rate for a fixed symbol rate) reduces degradation for channels with phase distortion, saturation, or adjacent channel interference unless the distortion is so severe that the hard decision error rate is extremely high.

• A simple technique is given for converting a convolutional/coder Viterbi decoder from rate-1/2 to rate-3/4 thereby increasing the data rate at a fixed symbol rate. Over the ideal PSK memoryless channel, the required E_b/N_o is increased by only 1 dB. The technique is to delete symbols from the rate-1/2 coder output and reintroduce them as erasures at the rate-1/2 decoder input. The best rate-3/4 codes generated by this technique achieve the maximum possible free distance for a given constraint length.

• Use of the LMS algorithm to control the tap weights of an adaptive equalizer gives a resulting QPSK performance essentially equal to the best possible (minimizing the bit error rate). Tap spacing does not need to be less than the bit duration.

• The decision-directed adaptation of an IF equalizer can be controlled from bit decision samples out of the modem. Even simpler, it is found that good equalization is achieved with control to minimize the alpha flunk output of the modem.

- Sequential dither control of the tap weights is a feasible adjustment strategy, and with error correction coding, control based on the decoder output has been found unnecessary (and would be incompatible with the use of alpha flunk).

- Adaptive equalization of phase distortion in the nonsaturating channel eliminates almost all degradation produced by the distortion.

- For a given peak level of phase distortion over the filtered data bandwidth, parabolic distortion produces far less degradation than sine or cosine (ripple-type) distortion, suggesting that the allowable limits for the different types of distortion should be specified separately.

- The phase distortions which can be corrected by the (manually) adjustable phase equalizer provided with the Comtech Frequency converters do not contribute a significant amount of performance degradation with FDMA. Conversely, these equalizers are not capable of correcting phase distortions which can severely degrade system performance.

- For FDMA operation, sharp cutoff filters are required in both the transmitter and receiver to permit close spacings of the adjacent signals, so as to achieve the maximum bandwidth utilization.

- The use of equalized filters (Figure 7-3) with bandpass equal to the bit rate for BPSK or half the total bit rate for QPSK improves system performance when an integrate-and-dump detector is used. This permits QPSK operation (uncoded) at ? s/Hz with minor degradation and, in the linear channel, allows FDMA signal spacing by as little as 0.55 times the total data rate, (for uncoded QPSK).

- Maximally flat filters (phase distortionless) are optimal if a sampling detector is used.

- With either the Phase II or the Phase III satellite TWT, use of equalized filters permit signal separations as close as 0.6 times the total data rate (for uncoded QPSK). Inclusion of a hard limiter in the earth terminal transmitter negates the advantages of the equalized filter, and minimum separation for FDMA is 0.75 times the total data rate (for uncoded QPSK).

- For FDMA operation at lower error rates with uncoded data, an optimum TWT output backoff exists at which the total system loss (sum of output backoff from saturation plus degradation) is minimized. For FDMA operation at higher error rates or with rate-1/2 error correction, the TWT should be operated at saturation.

- At the lower backoff levels, the Phase III TWT produces somewhat higher degradation to SQPSK/FDMA signals than the Phase II TWT.

- Degradation to uncoded QPSK/FDMA or SQPSK/FDMA signals due to AM/PM conversion becomes severe for a conversion coefficient above 6 degrees/dB.

- Rate-1/2 error correction coding of the FDMA signals and use of equalized filters reduces degradation in the saturating channel and permits signal spacings of 1.1 times the data rate. Degradation is also reduced by the error correction coding when the earth terminal transmitters hard limit, and feasible signal separation then is 1.3 times the data rate.

- For uncoded QPSK/TDMA or SQPSK/TDMA operation, the satellite TWT should be operated at saturation to minimize the total system loss.

- Narrowband hard limiters in the transmitter and/or satellite produce somewhat more degradation to SQPSK than to QPSK. Bandpass as narrow as 0.6 times the total bit rate is possible, but degradation decreases as the bandwidths increase.

- Duobinary signaling can increase the bandwidth utilization by 50 percent to 3 bps/Hz with no penalty in E_b/N_o over QPSK with conventional demodulation, assuming use of the Viterbi algorithm (a simple two-state algorithm) for maximum-likelihood demodulation of the duobinary and an equalizer to produce the necessary duobinary channel response.

- The use of rate-1/2 coding for duobinary produces a coding gain comparable to that achieved in the memoryless channel. However, the rate-1/2 Viterbi decoder for the duobinary channel requires twice the number of states as the decoder for the memoryless channel as a consequence of the duobinary intersymbol interference, and is quite different from the standard Viterbi decoder for the memoryless channel.

- Despite the amplitude variation, duobinary signaling is compatible with a saturating channel, with a small degradation in E_b/N_o.

10.2 RECOMMENDATIONS

The following recommendations are made for additional work.

- The duobinary technique appears particularly attractive for improved bandwidth utilization with either QPSK/FDMA or QPSK/TDMA. A study of the degra-

dation to duobinary caused by channel distortions, similar to that described in the report for conventional modulation, should be carried out via the computer simulation approach. The feasibility of using high rate error correction coding with duobinary should be studied; e.g., rate-3/4 coding enables 2.25 bps/Hz. This recommended study should be practically oriented to the DSCS application and include implementation considerations.

      ● The IF adjustable equalizer design based on a coaxial cable delay line implementation should be fabricated in a feasibility model for 10 Mbps data rate. This will enable the demonstration of equalizer control from the alpha flunk output of the QPSK modem, the correction of severe channel phase distortion, and the improved performance from equalization of a narrow bandwidth channel.

APPENDIX A
SOFTWARE DESCRIPTION OF SIMULATION PROGRAM SIMA

This appendix covers software considerations of the computer program
SIMA. The technical description of the operation and capabilities of SIMA are
given in Section 3. The following contains a description of all important system
parameters, all input parameters, and all common statements in the program. In
addition, descriptive writeups of the various subprograms called by SIMA are
included.

An overall flow diagram of the main program is given in Figure A-1 and
an illustration of the data input and computer output are given in Figure A-2 to guide
the user. Note that program SIMA is written in the FORTRAN IV language except for
a small number of subprograms written in assembly language for efficiency, and the
entire program has been developed on an IBM 370/135 computer system. The
program is suited for either batch or timeshare operation.

A.1   INPUT PARAMETERS OF PROGRAM SIMA

To provide flexibility of application, program SIMA provides for a
multiplicity of input parameters. These inputs permit structuring of the program
flow to simulate a variety of system configurations and allows setup of system and
element parameters to the desired values. All inputs are performed via NAMELIST
statements so that any combination of parameters may be changed from default
values in a given run without the necessity of entering those parameters whose values
are not to be changed.

Three namelist statements exist. Namelist INPT controls the main
program flow and contains all system parameters except the individual filter
parameters and the convolutional encoder/Viterbi decoder parameters. Namelist
FLTR is called at each filter location to permit the user to enter the filter parameters

INITIALIZE PARAMETERS TO DEFAULT VALUES

READ NAMELIST /INPT/

A

IRUN

0 → **QUIT**

1 → **RE-INITIALIZE ARRAYS AND PARAMETERS**

DO 20 I = 1, CHAN

CALL 'XCOD' TO SET UP PN SEQUENCES (CODED OR UNCODED) FOR CHANNEL I

'CMOD' PHASE MODULATES THE PN SEQUENCES

READ /FLTR/

'PPIN' SETS UP AND FILTERS CHANNEL **

HARD-LIMIT CHANNEL

TRANSLATE CHANNEL TO DESIRED CENTER FREQUENCY **

SET UP THE "NCHAN" CHANNELS

** OPERATION PERFORMED IN THE FREQUENCY DOMAIN (OTHER OPERATIONS IN THE TIME DOMAIN)

776-3597
(UNCLASSIFIED)

Figure A-1. Flow Diagram of Program SIMA

A-2

SET UP
THE
"NCHAN"
CHANNELS

ADD NEW CHANNEL
TO EXISTING
CHANNELS **

20

MDIST   0   1, 2, 3, 4, 5

'DIST' ADDS
CHANNEL PHASE
DISTORTION **

LOCB   0   1

READ/FLTR/

'PPIN' SETS UP AND
FILTERS CHANNEL **

MAMP   0   1

CALL 'AMP' TO
PERFORM
AMPLIFICATION

A-3

TRANSMITTER ↕ RECEIVER

776-3598
(UNCLASSIFIED)

A-4

A-5
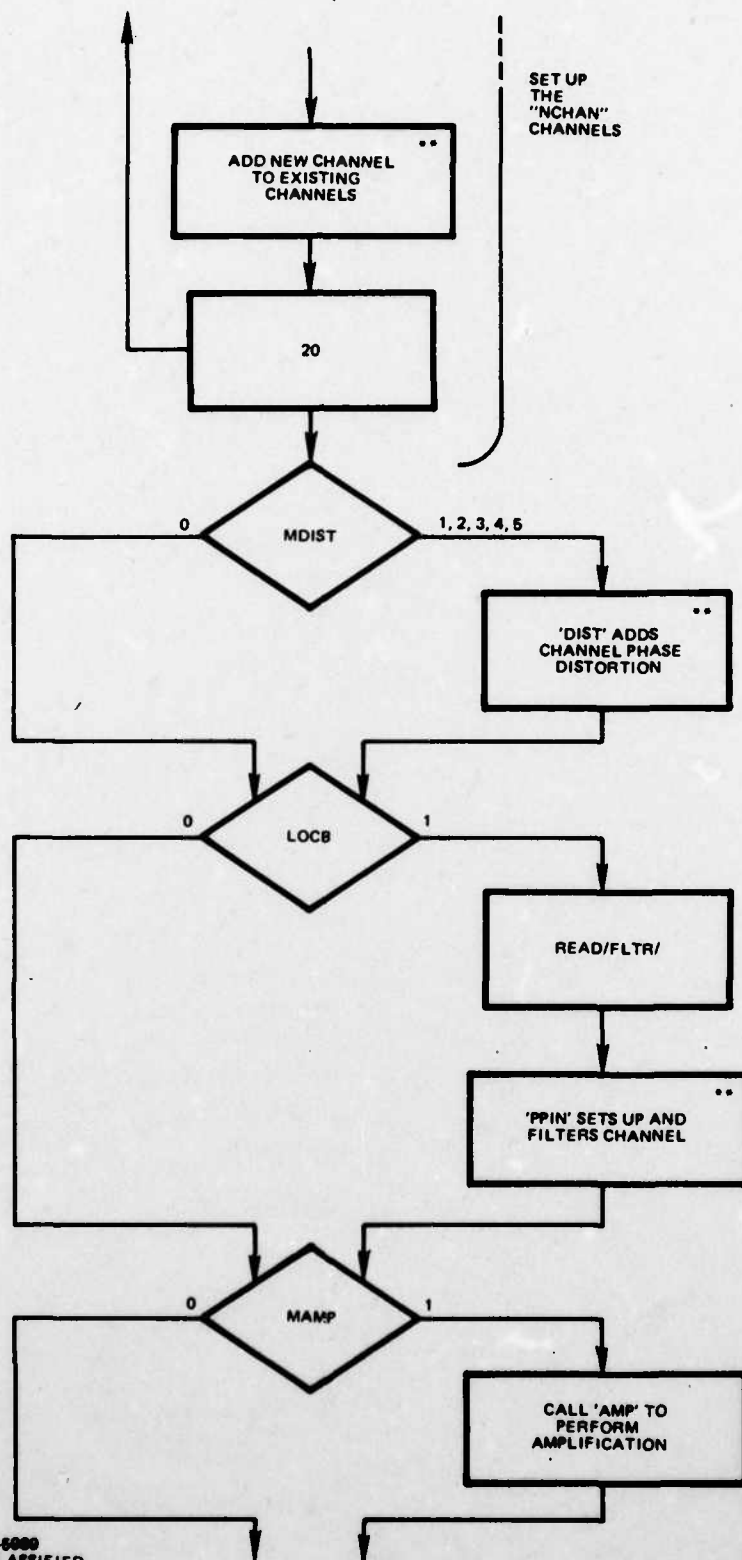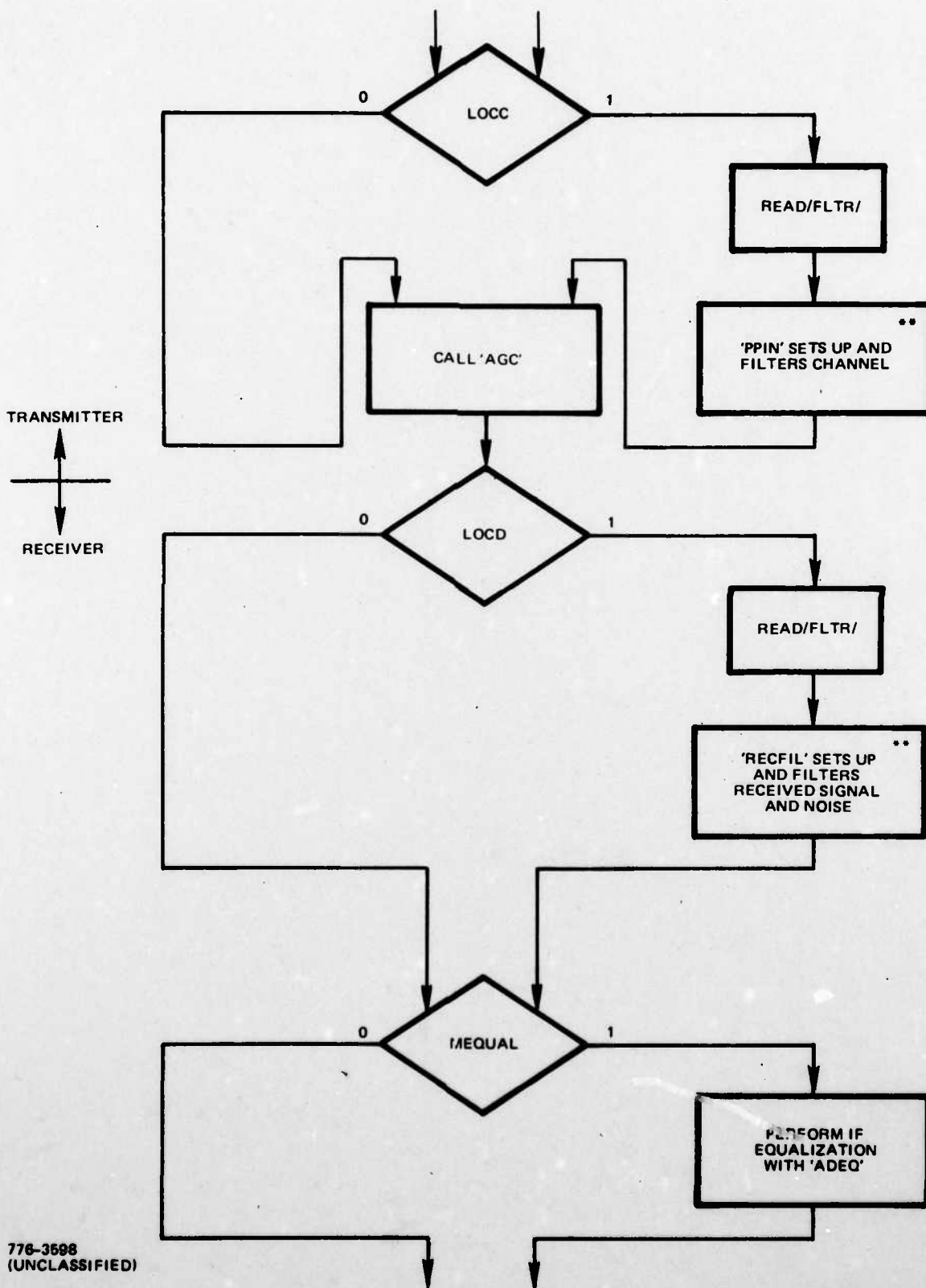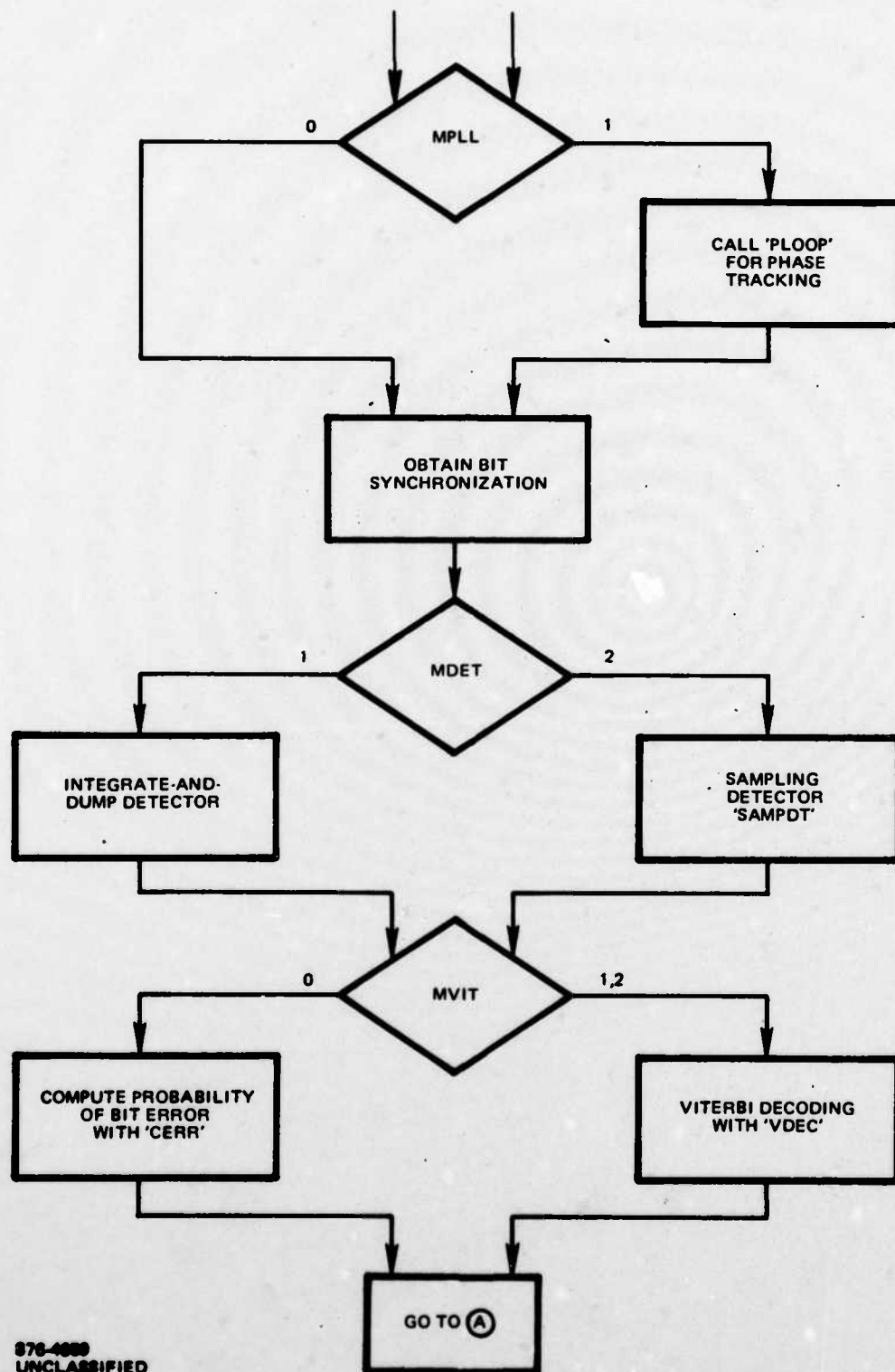
```
 1    &inpt loca=1,snr1=4.,snr2=10.,dsnr=3.,&end
 2    ENTER PARAMETERS FOR CHANNEL NO. 1
 3    &fltr mflt=1,bt=1.,np=5,&end

 4    OPTIMUM SYNCH=0.5469 BAUDS. AGCC=1.3435. LOOP=    0.493 DEGREES
 5    SNR= 4.0 DB:   PBEI=   0.172E-01 ( 0.50 DB.)     PBEQ=   0.173E-01 ( 0.52 DB.)
 6    SNR= 7.0 DB:   PBEI=   0.157E-02 ( 0.60 DB.)     PBEQ=   0.161E-02 ( 0.63 DB.)
 7    SNR=10.0 DB:   PBEI=   0.217E-04 ( 0.78 DB.)     PBEQ=   0.229E-04 ( 0.81 DB.)

 8    &inpt locd=1,snr1=10.,&end
 9    ENTER PARAMETERS FOR CHANNEL NO. 1
10    &fltr &end
11    ENTER PARAMETERS FOR FILTER NO. 4
12    &fltr &end

13    OPTIMUM SYNCH=0.0781 BAUDS. AGCC=1.3401. LOOP=   1.046 DEGREES
14    SNR=10.0 DB:   PBEI=   0.160E-04 ( 0.63 DB.)     PBEQ=   0.166E-04 ( 0.65 DB.)

15    &inpt lrun=0,&end
```

Figure A-2. Sample Run of Program SIMA

used there. Namelist IDEC is called if channel coding is specified and permits control of the encoder/decoder parameters.

This section lists the input parameters of each namelist together with the default value of the parameter. The default value is that value used by the program in lieu of a different value specified by the user. Inputs are entered via the appropriate namelist statement. Any number of the possible parameters may be changed on any input call. The last value given to a parameter is used for subsequent runs until changed by user specification of a new value. The format for entering data via the namelist INPT call is

b & INPT b P1 = xx, P2 = xx, P8 = xx, P5 = xx, & END

where b designates a space, & is the ampersand character, P represents a parameter name, and xx represents the numerical value assigned to that parameter. Value types must match the variable type (integer, real, complex, etc.). The parameters to be changed may be input in any sequence. The corresponding namelists FLTR and IDEC are handled with the same format. Following the last run, the input

b & INPT b IRUN = 0, & END

terminates execution of the program.

	PARAMETER NAME	DESCRIPTION	TYPE*	UNITS	DEFAULT VALUE
Accessing Signal Specification	MPSK	(=2) biphase; (=4) quadriphase	I	---	4
	NSMP	Number of sampling points	I	---	1024
	LSEQ	Length of PN sequences used	I	---	32
	QDEL	Displacement of quadrature data from in-phase data	R	Bits	0
	ISQ	(=0) quadriphase; (=1) staggered quadriphase	I	---	0
	NCHAN	Number of channels through transmitter	I	---	1
	CPR	Array specifying power of each channel (CPR (1) is center signal power)	R	---	1
Channel Filter Locations	LOCA	Sets up channels, filters and frequency offsets each as desired (=0) no set up; (=1) set up the NCHAN channels	I	---	1
	LOCB	Indicates channel filter preceding nonlinear amplifier; (=0) no filter; (=1) filter	I	---	0
	LOCC	Indicates channel filter following non-linear amplifier; (=0) no filter; (=1) filter	I	---	0
	LOCD	Indicates filter in receiver (=0) no filter ; (=1) filter	I	---	0
System Nonlinearities	LIM	Transmitter hard limiter; (=0) on; (=1) off	I	---	0
	MAMP	(=0) no amplifier; (=1) amplifier	I	---	0
	TYPE	(=0) linear amplification; (=1) TWT; (=2) soft limiter; (=3) hard limiter; (=4) amplitude characteristic of a Phase III TWT (=5) amplitude characteristic of Phase II TWT	I	---	0
	BO	TWT backoff (absolute value)	R	dB	3
	GAIN	Amplifier gain (optional entry)	R	dB	3

*I = Integer, R = Real.

	PARAMETER NAME	DESCRIPTION	TYPE*	UNITS	DEFAULT VALUE
Channel Phase Distortion	PHASE	(=0) no phase shift; (=1) Berman-Mahle; (=2) linear phase; (=3) truncated linear phase; (=4) AM-PM characteristic of a Phase III TWT; (=5) AM-PM characteristic of Phase II TWT	I	----	0
	PHS	Phase shift (for PHASE = 2 or 3)	R	Degrees/dB	0
	MDIST	(=0) no phase distortion; (=1) sin phase; (=2) cos phase; (=3) parabolic phase; (=4) sym step phase; (=5) non-sym step phase	I	----	0
	SK	Peak phase error (see text)	R	Degrees	5
	FR	Normalized ripple frequency (see text)	R	----	1
Phase Tracking	MPLL	(=0) no phase tracking; (=1) phase tracking; (=2) fixed phase rotation	I	----	1
	DEG	Fixed phase rotation (for MPLL = 2)	R	Degrees	0
	BPL	Loop filter bandwidth	R	Hertz	0.01
	NORD	Loop order (first, second or third)	I	Order	2
	LOOP	(=0) phase locked loop; (=1) Costas loop	I	----	1
AGC	MAGC	Indicates mode of automatic gain control (=0) normal automatic operation (=1) uses value AGCC, AGC defeated	I	----	0
	AGCC	Value of AGC used when MAGC = 1	R	----	1
Bit Sync	MSYN	Indicates mode of bit sync loop (=0) normal automatic operation; (=1) uses value XDEL, sync loop defeated	I	----	0
	XDEL	Value of sync delay used when MSYN = 1	R	----	0

*I = Integer, R = Real.

	PARAMETER NAME	DESCRIPTION	TYPE*	UNITS	DEFAULT VALUE
Adaptive Equalizer	MEQUAL	Indicates IF equalizer in receiver (=0) no equalizer; (=1) adaptive equalizer	I	---	0
	NTAP	Number of equalizer taps	I	---	32
	MREF	Mode of obtaining desired response (=0) training sequence; (=1) decision feedback	I	---	0
	ISP	Tap spacing in number of signal samples per tap	I	---	8
	ISHIFT	Relative displacement between input signal samples and desired response samples (in number of samples)	I	---	0
Detector	MDET	Type of bit detector in receiver (=1) integrate-and-dump detector (=2) sampling detector	I	---	1
	SNR1	First value of signal-to-noise ratio	R	dB	1
	SNR2	Last value of signal-to-noise ratio	R	dB	15
	DSNR	Step size of signal-to-noise ratio	R	dB	1
Convolutional Coder/Viterbi Decoder	MVIT	Mode of channel coding used (=0) coding not in operation; (=1) convolutional encoding, Viterbi decoding (serial mode); (=2) convolutional encoding, Viterbi decoding (parallel mode)	I	---	0
	KON	Convolutional encoder/decoder constraint length	I	---	5
	MQUAN	Inputs to Viterbi decoder; (=0) unquantized soft decisions; (=1) ideal 3-bit quantized; (=2) α-flunk 3-bit quantized	I	---	0
	IDISP	Mode determining spectral plotting (=0) no plots; (=1) plot data	I	---	0
	IRUN	Run execution indicator (=0) terminate execution; (=1) run next case	I	---	1

*I = Integer, R = Rate.

A.3   INPUT PARAMETERS FOR NAMELIST FLTR

PARAMETER NAME	DESCRIPTION	TYPE*	UNITS	DEFAULT VALUE
MFILT	Type of filter used (=0) no filter; (=1) Butterworth; (=2) Chebyshev; (=3) Gaussian; (=5) phase distortionless Butterworth; (=6) equalized filter response	I	---	0
BT	One-sided filter bandwidth (normalized to bit rate)	R	---	1
NP	Number of filter poles ($1 \leq NP \leq 15$)	I	---	5
RIP	Chebyshev filter ripple amplitude	R	dB	.05
FCENT	Filter center frequency offset (normalized to bit rate)	R	freq. offset/ bit rate	0

*I = Integer, R = Rate.

A.4   INPUT PARAMETERS FOR NAMELIST IDEC

PARAMETER NAME	DESCRIPTION	TYPE*	UNITS	DEFAULT VALUE
ILAST	Controls number of passes through decoder. For each run, the parameters of the decoder can be changed (=0) terminate decoder run; (=1) set up another run through decoder	I	---	0
IRATE	Number of bits per node (inverse of code rate)	I	---	2
SNR	E_b/N_0 at which probability of bit error is evaluated	R	dB	3
KNT	Number of bits through decoder for computing error rate. (KNT should be at least 10 divided by the expected error rate at given SNR.)	I	---	10,000

*I = Integer, R = Real.

A.5   PARAMETERS IN COMMON STATEMENTS USED IN SIMA

A.6   COMMON/PAMP/PHASE, TYPE, GAIN, BO, PHS, VO, GI

PHASE = phase characteristic of nonlinear amplifier

TYPE = amplitude characteristic of nonlinear amplifier

GAIN = gain slope of amplifier

BO = TWT backoff value in decibels

PHS = number of degrees per dB value for amplifier phase characteristic

VO = internally computed TWT output voltage at given backoff

GI = internally computed scaling of amplifier input to produce the desired backoff.

A.7   COMMON/FLT/FNC, PP(15, 2), WC, NP, MFILT, RIP

FNC* = normalization factor for filters to produce unity amplitude response at filter center frequency

PP*(15, 2) = poles and residues of filter

WC = 3 dB filter bandwidth, in radians

NP = number of filter poles

MFILT = type of filter (Butterworth, Chebyshev, or Gaussian)

RIP = Chebyshev filter ripple amplitude, in decibels.

A.8   COMMON/SIG/TBD, MDIST, P(4), G(4), SK, FR, LSEQ

TBD = chip time (normally = 1)

MDIST = type of phase distortion in channel

P(4) = four values of modulator phase (degrees)

G(4) = four values of modulator gain (decibels)

SK = peak channel phase error (degrees)

FR = normalized channel phase error frequency (see text)

LSEQ = number of chips in PN sequence.

A.9   COMMON/PLL/V(3), WW(3), E(3)

V(3) = Array for phase tracking loop storing frequency values

W(3) = array storing internally computed loop coefficients

E(3) = array storing output error voltages for loop.

*Implies parameter ... ex.

A-13

A. 10   COMMON/DAT/Z, NSMP, IDELB, ANSB, J1

Z(1024) - complex array of signal samples

NSMP - number of samples in array z

IDELB - number of samples delay of data in quadrature channel from data
         in in-phase channel (for QPSK)

ANSB - number of samples per bit in array Z

J1 - optimum bit sync offset (in samples)

A. 11   COMMON/FFT/M, INV, S

M(3), INV(256), S(256) explained in IBM writeup of Fast Fourier
Transform HARM

A. 12   COMMON/DCD/IRATE, IMP, ISTATE, K, NTAP, LOGIC, AMET, IPATH

IRATE = number of bits per node

K = constraint length

ISTATE = current state of encoder

IMP = encoder interleaved impulse response

NTAP(4) = tap vectors of encoder/decoder

LOGIC(256) = array of expected bit polarities in Viterbi decoder

AMET(256) = array of metrics generated in Viterbi decoder

IPATH(256) = array of path states generated in Viterbi decoder.

A. 13   COMMON/SWP/IMP(16), LO(1024), AM(1024), IP(1024)

COMMON/SWP/ is used in subroutine SWAP to store and exchange the
various Viterbi decoder parameters and metrics in COMMON/DCD/ between various
channels as required.

A. 14   DESCRIPTION OF SUBPROGRAMS CALLED BY SIMA

This section contains subroutine and function descriptions to enable under-
standing of their purpose and usage. Pertinent parameters are described and sub-
programs called are listed. Remarks are given when required concerning considerations
or limitations involved when using the routine. The technical description of the functions
performed in these subroutines is given in Section 3.

SUBROUTINE: ADEQ

PURPOSE

ADEQ simulates an IF adaptive equalizer using the LMS algorithm for adaptation. ADEQ can be used in a training sequence mode or with decision feedback based on hard-limiting of the equalizer output samples to obtain the reference.

USAGE

Call ADEQ (X, D, NSMP, NTAP, ISP, PWRN, MREF, ISHIFT). Calling program must contain the following statements COMPLEX X(NSMP), D(NSMP).

DESCRIPTION OF PARAMETERS

- X – array of input time samples, X is returned as array of equalizer array of equalizer output samples.
- D – array containing training sequence of output samples from the ideal transmitter phase modulator.
- NSMP – number of samples in arrays X and D.
- NTAP – number of tap weights in equalizer.
- ISP – spacing of tap weights (i.e., number of samples per tap).
- PWRN – internally generated scaling of noise power.
- MREF – determines decision mode of equalizer; (=0) training sequence using array O; (=1) decision-directed using hard-limited equalizer output.
- ISHIFT – permits a displacement of the alignment of input samples and training sequence samples by ISHIFT samples. Normally, ISHIFT = 0.
- W – array of (complex) tap weights generated.

REMARKS

ADEQ adapts by iterating through the NSMP input time samples a total of 12 times, using a convergence constant of .001 to insure stability. The resulting mean square error between the NSMP output samples X and desired response samples D is printed together with the final tap weights W and the output noise energy PWRN. White Gaussian noise of unit energy is assumed at the input of the equalizer and PWRN represents the noise energy after passage through the equalizer and the integrate-and-dump filter.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

HARM.

SUBROUTINE: AMP

PURPOSE

AMP simulates several nonlinear amplifiers, including both amplitude and phase effects on the time-sampled input signal, by calls to function NLN. AMP can model hard and soft limiters, Travelling Wave Tube (TWT) amplifiers, as well as simulating only the phase or only the amplitude characteristics of such amplifiers.

USAGE

Call AMP(X, NSMP). Calling program must include the following statements COMMON/PAMP/-, EXTERNAL OPPT, COMPLEX X(NSMP).

DESCRIPTION OF PARAMETERS

TYPE – type of nonlinear amplifier amplitude characteristic to be modelled; (=0) linear amplification; (=1) Berman-Mahle TWT model; (=2) soft limiter; (=3) hard limiter; (=4) amplitude characteristic of a Hughes' TWT; (=5) amplitude characteristic of the Phase II TWT.

PHASE – form of nonlinear amplifier phase shift characteristic to be modelled; (=0) no phase shift; (=1) Berman-Mahle phase characteristic; (=2) linear phase shift, in degrees per dB; (=3) truncated linear phase shift; (=4) AM-PM characteristic of a Hughes' TWT; (=5) AM-PM characteristic of Phase II TWT.

PHS – phase shift in degrees per dB (for TYPE = 2).

GAIN – gain slope of amplifier.

BO – backoff of TWT from maximum output power (dB).

VO – internally computed output voltage at given backoff.

GI – internally computed scaling of input signal to produce the desired backoff.

REMARKS

Subroutine AMP accepts the NSMP time samples X of the input signal and scales the amplitude and phase of each sample to simulate the action of the nonlinear amplifier being modelled. AMP returns the array X of output samples to the calling program.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

NLN, AGC, ADB, OPPT, NRR, FMIN.

SUBROUTINE: CERR

PURPOSE

CERR accepts an array of detector output quantities (for both the I and Q channels) and determines the resulting probability of error and system degradation at user specified signal-to-noise ratios.

USAGE

Call CERR (LSEQ, PWRN, SNR1, SNR2, DSNR). Calling program must contain the following statements COMMON/DCD/-.

DESCRIPTION OF PARAMETERS

LSEQ - number of bits in the input PN sequence.

SNR1 - first signal-to-noise ratio at which probability of bit error is computed (in dB).

SNR2 - last signal-to-noise ratio at which probability of bit error is computed (in dB).

DSNR - step size between SNR, in dB.

PWRN - precomputed change in input noise energy due to noise passage through receiver filters and/or equalizer.

REMARKS

CERR accepts detected bits through the COMMON array SUM. If no noise filtering is performed, PWRN should be initialized to unity.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

AWERC, ERR.

FUNCTION: CMOD

PURPOSE

CMOD is used to phase modulate an input time sample X. CMOD can model an ideal biphase or quadriphase modulator and simulate a phase modulator with amplitude and phase errors.

USAGE

Y = CMOD(X, J). Calling program must contain the following statements COMMON/FLT/-;COMMON/SIG/-;COMPLEX CMOD, Y.

DESCRIPTION OF PARAMETERS

X - input (real) time sample to be phase modulated.

J - channel in which sample X is to be placed; (=0) I channel; (=2) Q channel.

P - array of four phase locations (nominally 0°, 90°, 180°, 270°).

G - array of four amplitude values (nominally 0 dB). (Note that P and G can be externally changed to other values.)

SUBROUTINE: CORD

PURPOSE

Performs node synchronization for Viterbi decoder.

USAGE

Call CORD (LSEQ, SUM, RI, RQ).

DESCRIPTION OF PARAMETERS

LSEQ - number of bits in PN sequence.

SUM - array of I and Q soft decisions from detector.

RI - encoded bit sequence for in-phase (I) channel.

RQ - encoded bit sequence for quadrature (Q) channel.

REMARKS

Subroutine XCOD performs a cross-correlation of the soft decisions and the encoded bit sequences to correct for channel delay and interchange of I and Q data prior to Viterbi decoding.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

None.

SUBROUTINE: CORNS

PURPOSE

CORNS develops correlated noise samples for input to the Viterbi decoder to account for the presence of receiver filters and/or equalization in the receiver.

USAGE

Call CORNS (K, AN, BN, IR). Calling program must contain the following statements: COMMON/FFT/-   and COMMON/DAT/-

DESCRIPTION OF PARAMETERS

K – normally = 0, set = 1 at the beginning of a new set of LSEQ input samples.

AN – in-phase noise sample returned to calling program.

BN – quadrature noise sample returned to calling program.

IR – parameter controlling Gaussian generating program SNORM. Initialize IR = 9 at first call to CORNS.

REMARKS

If K = 0, CORNS sets up NSMP correlated noise samples based on the composite receiver frequency response stored in complex array DDD. Thereafter, each call returns a successive noise sample based on the integrate-and-dump output.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

SNORM, HARM.

SUBROUTINE: DISPLA

PURPOSE

DISPLA accepts an array of NSMP frequency samples and generates all calls for a CALCOMP plotter.

USAGE

Call DISPLA (Z, IPL).

DESCRIPTION OF PARAMETERS

Z – array of NSMP complex frequency samples to be plotted.

IPL – parameter initializing plotter. IPL = 0 on first call to DISPLA in program; IPL = 1 thereafter.

SUBROUTINES AND PROGRAM FUNCTIONS REQUIRED

Normal calls to CALCOMP plotting subroutines.

FUNCTION: DIST

PURPOSE

DIST models several forms of phase distortion in a communications channel, specifically sinusoidal, cosinusoidal and parabolic phase variation with frequency and step changes in phase at specified frequencies.

USAGE

Y = DIST(S). Calling program must contain the following statements COMPLEX DIST, Y, S; COMMON/FLT/-;COMMON/SIG/-.

DESCRIPTION OF PARAMETERS

S – complex input frequency samples (radians/sec).

MDIST – type of phase distortion modelled; (=0) no distortion; (=1) sinusoidal phase variation with frequency; (=2) cosinusoidal phase variation with frequency; (=3) parabolic phase variation with frequency; (=4) step phase change at frequency FR.

FR – normalizes ripple frequency (see text).

SK – peak phase error for MDIST = 1 or 2 phase error at frequency FR for MDIST = 3 magnitude of phase step at frequency FR for MDIST = 4.

SUBROUTINE: RECFIL

PURPOSE

RECFIL is called at the location of each receiver filter to determine the reduction of noise energy produced by that filter. White Gaussian channel noise is assumed present at the receiver input.

USAGE

Call RECFIL (CN, MFILT, DDD).

DESCRIPTION OF PARAMETERS

CN – ratio of noise energy at filter output to that present at filter input.
MFILT – receiver filter type.
DDD – array containing frequency response of filter.

SUBROUTINE: SAMPDT

PURPOSE

SAMPDT models a sampling detector. The subroutine determines the optimum sampling time for the input bit sequence to produce the minimum probability of bit error.

USAGE

Call SAMPDT (Z, NSMP, NSB, QDEL, LSEQ, J1). Calling program must contain the following statements COMPLEX Z(NSMP); COMMON/DCD/-.

DESCRIPTION OF PARAMETERS

Z – complex array of time samples into detector.

NSMP – number of time samples used.

NSB – number of time samples per bit.

QDEL – relative delay of bit transitions between I and Q channels (given in fraction of a bit).

LSEQ – length of PN sequence in I and Q channels.

J1 – estimated filter delay (fraction of a bit) to assist internal optimization of sampling time.

REMARKS

SAMPDT returns the detected samples in array SUM(64, 2) to calling program. Subroutine CERR uses these samples to compute the probability of bit error.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

WERFC.

SUBROUTINE: VDEC

PURPOSE

Subroutine VDEC performs all functions required to set up and emulate the Viterbi decoder structure by calling the appropriate subroutines.

USAGE

Call VDEC (ISEQ, KSEQ, MVIT, MEQUAL, MQUAN). Calling program must contain the following statements COMMON/DCD/-; COMMON/SWP/-; DIMENSION ISEQ(32), KSEQ(32). The following namelist is read from VDEC: NAMELIST/IDEC/ K, ILAST, IRATE, SNR, IRUN, KNT.

DESCRIPTION OF PARAMETERS

IRATE — number of bits per mode.

K — constraint length.

ISEQ — transmitted bit sequence of I channel.

KSEQ — transmitted bit sequence of Q channel.

MVIT — mode of Viterbi decoder; (=0) no decoder; (=1) decode I and Q channels independently; (=2) decode interleaved I and Q channel data.

MEQUAL — (=0) if no filter or equalizer in receiver; (=1) otherwise.

MQUAN — (=0) if unquantized soft decisions to be decoded; (=1) provides 3-bit quantized soft decisions to decoder.

SNR — signal-to-noise ratio (in decibels) at which probability of error is to be computed.

KNT — number of channel output bits passed through decoder (should be at least ten times the expected number of channel errors occuring at given SNR).

IRUN — integer initializing noise function SNORM.

ILAST — integer determining number of passes through Viterbi decoder. For each pass, namelist /IDEC/ is read permitting changes in decoder parameters. (=0) no further passes; (=1) set up parameters for another pass.

REMARKS

VDEC stores the input samples to the decoder, permitting the simulation of decoders with different parameters or different KNT and IRUN values before returning to the calling program. Input bits to VDEC are obtained from the precomputed array SUM.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

SWAP, INDEC, SNORM, DECODE, CORNS, AWERFC.

SUBROUTINE: VITERB

PURPOSE

To simulate a convolutional encoder and Viterbi decoder.

USAGE

Call INCODR, INDEC (IDELAY), ENCODE (IIN, IOUT), DECODE (SOFT, IBIT). Calling program must contain the following statements COMMON/DCD/IRATE, IMP, ISTATE, K, NTAP (4), LOGIC(256), AMET(256), IPATH(256). EXTERNAL VITERB. Parameters IOUT and SOFT are arrays of dimension = IRATE.

DESCRIPTION OF PARAMETERS

IRATE - number of bits per node.

K - constraint length.

IDELAY - number of bits of path delay in final decision.

NTAP - tap vectors of encoder/decoder.

IIN - input bit (0 or 1) to convolutional encoder.

IOUT - array of encoded output bits.

*IMP - encoder interleaved impulse response.

*ISTATE - current state of encoder.

SOFT - array of soft decisions input to Viterbi decoder.

IBIT - decoded bit from decoder.

*LOGIC - internally generated array of expected bit polarities.

*AMET - internally generated array of metrics.

*IPATH - internally generated array of path states.

REMARKS

Subroutine INCODR is called to initialize the encoder. Subroutine INDEC is called to initialize the decoder. Following initialization, subroutine ENCODE is called to encode the data while subroutine DECODE is called to decode the data. Note that the decoded bits lag behind bits input to the decoder by an amount = IDELAY + K - 1.

Note that parameters with an asterisk above are internally generated within VITERB but are available for external observation and/or manipulation if desired. Both the encoder and decoder assume a shift register which shifts bits to the left. The leading and trailing bits of each tap vector (NTAP) must be unity. The constraint length (K) must be of value 3 to 8 and the number of bits per node (IRATE) can be no

VITERB (cont.)

larger than 4.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

IAND, IXOR.

SUBROUTINE: XCOD

PURPOSE

Generation of PN sequences for biphase and quadriphase channels. Also calls convolutional encoder when required.

USAGE

Call XCOD (RI, RQ, ISEQ, KSGQ, MVIT, LSEQ, ISTAT, JS, IJS, ITAP). Calling program must contain the following statements COMMON/DCD/-;EXTERNAL VITERB; DIMENSION RI(64), RQ(64), ISEQ(32), KSEQ(32).

DESCRIPTION OF PARAMETERS

ISEQ – PN bit sequence for in-phase data.

KSEQ – PN bit sequence for quadrature data.

RI – encoded bit sequence for in-phase channel.

RQ – encoded bit sequence for quadrature channel.

MVIT – Form of encoding: (1) codes I and Q channels independently;
(2) interleaves coded data between I and Q channel for rate = $1/2$ codes.

LSEQ – number of bits in PN sequence.

ISTAT – current state of PN sequence generator.

JS – number of bits in maximal length PN sequence.

IJS – parameter controlling flow of XCOD.

ITAP – tap vector of PN sequence generator.

IIN – input bit (0 or 1) to convolutional encoder.

REMARKS

Subroutine XCOD is called to set up independent PN bit sequences for the I and Q channels of each signal in the system. In addition, if channel coding is employed, subroutine ENCODE is called to convolutionally encode these bits. If MVIT = 0, no encoding is performed. If MVIT = 1 is used, the I and Q channels are encoded independently. If MVIT = 2 and IRATE = 2, only one PN is used and for each input bit, the two resulting encoded bits are interleaved with one in the I channel and the second in the Q channel. Note that internal dimension limits the value of the constraint length (K) to be from 3 to 8 and the number of bits per node (IRATE) can be no larger than 4.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

MSRG, VITERB, IAND, IXOR, SNORM.

SUBROUTINE: PLOOP

PURPOSE

To simulate a phase-locked loop or a Costas loop with bit rate equal to sampling rate. Loop order = 1, 2, 3.

USAGE

Call PLOOPS (N, BL, W, V, E, PHASE); Call PLOOPR (EIN, IBIT, W, V, E, PHASE, N).

DESCRIPTION OF PARAMETERS

N – integer input = 1, 2, or 3, denoting loop order.

BL – floating point value input = loop noise bandwidth (one-sided) x sampling interval.

W(3) – floating point array output storing computed loop coefficients.

V(3) – floating point array storing frequency values, PLOOPS initializes to zero as output. For other initial state to PLOOPR set

   N = 2   V(3) =   (rad/sec) x sampling interval

   N = 2, 3, V(2) = V(3) – (rad/sec^2) x (sampling interval)^2

E(3) – floating point array output storing error voltages initialized to zero by PLOOPS

PHASE – floating point value denoting phase in radians initialized to zero by PLOOPS as output. Set for other initial state as input to PLOOPR.

EIN – complex value input = input voltage sample.

IBIT – integer input = 0 for PLL, 1 for Costas.

REMARKS

First initialize each loop with PLOOPS. (This computes W and initializes V, E, and PHASE to zero.) Set V and PHASE for other initial conditions to PLOOPR, which iterates loop.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

None.

SUBROUTINE: MSRG

PURPOSE

To simulate a modular shift-register binary code generator of any length to 32 bits. MSRG is in IBM assembly language.

USAGE

IOUT = MSRG(ITAP, ISTATE).

DESCRIPTION OF PARAMETERS

ITAP - (INTEGER *4) - Feedback tap code in reversed notation. The least-significant-bit of ITAP is the coefficient (0, 1) of the next-to-highest degree term of the polynomial. Coefficients of lesser degree terms are then placed in successively higher order bits of ITAP. Consequently, generators with less than 32 stages are specified by filling the unused most-significant-bit positions with zeros. The coefficient of the highest degree term is omitted.

ISTATE - (INTEGER *4) - Present state (i.e., contents) of the shift-register. The direction of shifting is from the most-significant-bit to the least-significant-bit. The new state is returned in ISTATE.

MSRG - (INTEGER *4) - The output of the shift register is returned as the integer value (0, 1) MSRG.

<div align="center">
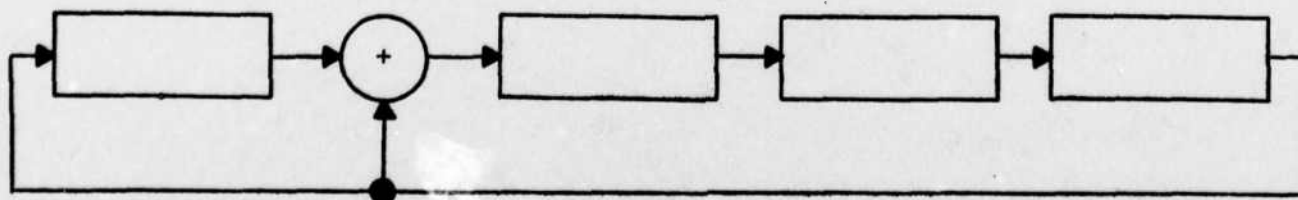
SUBROUTINE: SNORM

</div>

PURPOSE

To generate independent normally distributed samples. SNORM is in IBM assembly language.

USAGE

V = SNORM (IR).

DESCRIPTION OF PARAMETERS

IR – (INTEGER *4) – For the first entry, IR must contain a positive integer of the form $4n = 1$, n is an integer. Thereafter, IR will contain an integer generated by the subprogram for use on the next entry to the subprogram.

SNORM – (REAL *4) – The resultant normally distributed random number with zero mean and unit variance.

REMARKS

The method of generating the samples is algorithm FL_5 described in J. H. Ahrens and U. Dieter, "Extensions to Forsythe's Method for Random Sampling from the Normal Distribution", to appear.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

None.

SUBROUTINE: IXOR

PURPOSE

To perform exclusive ORs of two integer arguments, bit-by-bit. IXOR is in IBM assembly language.

USAGE

ISUM = IXOR (I1, I2).

DESCRIPTION OF PARAMETERS

I1 - (INTEGER *4) - First operand. It is returned unchanged.

I2 - (INTEGER *4) - Second operand. It is returned unchanged.

IXOR - (INTEGER *4) - The bit-by-bit modulo-two sum (exclusive OR) of the first and second argument is returned as the function value.

REMARKS

FORTRAN integer variables are represented in binary, two's-complement notation on the IBM 370 system.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

None.

SUBROUTINE: IAND

PURPOSE

To perform logical ANDs of two integer arguments, bit-by-bit. IAND is in IBM assembly language.

USAGE

IPROD = IAND (I1, I2).

DESCRIPTION OF PARAMETERS

I1 – (INTEGER *4) – First operand. It is returned unchanged.

I2 – (INTEGER *4) – Second operand. It is returned unchanged.

IAND – (INTEGER *4) – The bit-by-bit logical product (AND) of the first and second argument is returned as the function value.

REMARKS

FORTRAN integer variables are represented in binary, two's-complement notation on the IBM 370 system.

SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED

None.

ADDITIONAL SUBPROGRAMS USED

The following functions and subroutines are used to perform specific functional calculations and will be described in a briefer fashion than the preceding subprograms.

A.   FUNCTION: DB

PURPOSE

DB accepts a real quantity R and returns the value of R in decibels

$$S = 10 \log_{10} R$$

CALLING THE SUBPROGRAM

$S = DB(R)$.

B.   FUNCTION: ADB

PURPOSE

ADB accepts a quantity S expressed in decibels and returns the value of S as a real number

$$R = 10^{S/10}$$

CALLING THE SUBPROGRAM

$R = ADB(S)$.

C.   FUNCTION: WERFC

PURPOSE

WERFC accepts a quantity R and returns the Weber error function evaluated at R

$$S = \frac{1}{\sqrt{2\pi}} \int_R^\infty e^{-x^2/2} \, dx$$

CALLING THE SUBPROGRAM

$S = WERFC(R)$.

D.   FUNCTION: AWERFC

          PURPOSE

          AWERFC accepts a value of the Weber error function S and returns the
argument R of the error function.

          CALLING THE SUBPROGRAM

          $R = AWERFC(S)$.

E.   FUNCTION: AGC

          PURPOSE

          AGC accepts the array Z containing N complex numbers and returns the
rms value of these numbers.

          CALLING THE SUBPROGRAM

          $Y = AGC(Z, N)$.

F.   FUNCTION: FMIN

          PURPOSE

          Function FMIN finds the argument X of an external function F which
minimizes an external function F by performing a Golden Search over the interval
[X1, X2]. Parameter EPS determines the accuracy required in locating the exact minimum.

          CALLING THE SUBPROGRAM

          $X = FMIN(F, X1, X2, EPS)$; EXTERNAL F.

G.   COMPLEX FUNCTION: FFR

          PURPOSE

          FFR uses the poles computed for a filter to determine the complex filter
response at a frequency S.

          CALLING THE SUBPROGRAM

          $R = FFR(S)$; COMMON/FLT/-.

A-35

H. COMPLEX FUNCTION: FDOM

PURPOSE

FDOM accepts the complex array Z containing N frequency response samples and multiplies them by N values of an external function F evaluated at the same frequencies as the corresponding samples of Z. FDOM returns the array Z of multiplied samples.

CALLING THE SUBPROGRAM

Y = FDOM(F, Z, N); COMMON/SIG/.

I. SUBROUTINE: ERR

PURPOSE

ERR accepts an array SUM of L detector output values, computes the probability of error PER of each value at a given signal-to-noise ratio (not in decibels), and averages these L probabilities to return the average probability of error PER.

CALLING THE SUBPROGRAM

CALL ERR(PER, SUM, X, L).

J. SUBROUTINE: FBUT

PURPOSE

FBUT sets up the frequency response of a phase distortionless filter having a 5-pole Butterworth amplitude response. FBUT is called from FDOM if MFILT = 5.

K. SUBROUTINE: FCRC

PURPOSE

FCRC sets up the frequency response of the phase distortionless equalized filter described in Section 7.1. FCR is called from FDOM if MFILT = 5.

L. SUBROUTINE: SWAP

PURPOSE

Subroutine SQAP is called from Viterbi decoder program VDEC if multiple decoders are used. SWAP interchanges the various decoder parameters, paths, and metrics to provide VDEC with the appropriate decoder required.

A. 15   USING PROGRAM SIMA

Program SIMA has been constructed to facilitate user interaction in setting up the system to be simulated and for entering the specific program parameters desired. Multiple passes may be made through the program on each run of SIMA. As previously mentioned, namelist inputs are used throughout the program for data input. Only those parameters which are different from default values (or different from the previous pass) need be entered at each namelist call.

An example run with input commands and program output is shown in Figure 5. Table A-I explains each input or output line given in Figure 5. The first pass determines the degradation of a transmitter filter (LOCA) while the second pass evaluates the combined effects of identical transmitter (LOCA) and receiver (LOCD) filters. Note that a namelist/INPT/is called at the beginning of each pass or to terminate execution of SIMA. A namelist/FLTR/is called at each specified filter location. If Viterbi decoding is performed, namelist/IDEC/is called for each pass through the decoder. For the last pass through the decoder, set ILAST = 0.

Table A-I: Explanation of Sample Run Statements

Line numbers are given followed by an (I) for user input or an (O)
for system output. This is followed by an explanation of that state-
ment.

①	(I)	Specifies a transmit filter at location LOCA and degradation to be computed at input SNR = 4, 7, and 10 dB.
②	(O)	System warning for operator to enter filter parameters.
③	(I)	Filter parameters: 5-pole Butterworth filter with BT = 1.
④	(O)	Values of bit sync, AGC, and phase loop rotation.
⑤	(O)	I and Q error rates and system degradations at SNR = 4 dB.
⑥	(O)	I and Q error rates and system degradations at SNR = 7 dB.
⑦	(O)	I and Q error rates and system degradations at SNR = 10 dB.
⑧	(I)	Rerun with parameters as before but with receiver filter on (LOCD) and degradation computed only at SNR = 10 dB.
⑨	(O)	Warning: enter parameters for filter at LOCA.
⑩	(I)	Filter parameters unchanged.
⑪	(O)	Warning: enter parameters for filter at LOCD.
⑫	(I)	Filter parameters unchanged.
⑬	(O)	Value of bit sync, AGC, and phase loop rotation.
⑭	(O)	I and Q error rates and system degradations at SNR = 10 dB.
⑮	(I)	Terminate execution.

For single carrier operation, namelist/FLTR/is called at point LOCA. For NCHAN carriers, namelist/FLTR/is called NCHAN times at location LOCA to permit required filtering and frequency offset for each carrier. The last call to /FLTR/ at LOCA is the baseband carrier on which detection and decoding is performed. Thus, this last call to /FLTR/ should have FCENT = 0.

Several precautions must be considered for multiple carrier operation besides the multiple calls to /FLTR/ at location LOCA. First, AGC normalizes the total (composite) signal power to unity rather than just the baseband signal power. In addition, the bit sync loop can yield erroneous sync due to the multiplicity of signal crossings. To avoid these problems the following procedure can be used when running such cases:

a.   First, run the system with all elements present and only a single carrier. Determine the value of AGC and SYNC obtained as output.

b.   Next, run the multiple carriers through the system but defeat the AGC and bit sync loops by entering MAGC = 1 and MSYN = 1. Also enter the values of AGC and SYNC previously obtained by means of input parameters AGCC and XDEL. All of these parameters are input through the namelist/INPT/call.

Finally, a plot of the spectral response of the system at locations LOCA, LOCB, LOCC, and LOCD can be obtained by specifying the locations desired and entering IDISP = 1.

APPENDIX B
INTERMODULATION ANALYSES

Using analytical approaches, this appendix presents some computational results for intermodulation with multiple signals in a channel with saturation and AM/PM conversion. One analysis presumes two or three signals and computes intermodulation for AM/PM conversion. The second analysis is intended to treat the case of many signals in FDMA, and does so by making the assumption that a large number of signals becomes equivalent to a Gaussian noise process.

B.1   COMPUTATION OF INTERMODULATION PRODUCTS DUE TO AM/PM CONVERSION IN CHANNEL

AM/PM conversion causes intermodulation products to appear when several signals access the channel simultaneously. In this respect, AM/PM conversion has an effect similar to that of soft limiting where the envelope input/output characteristic is nonlinear without an accompanying phase shift.

We assume here a general input/output characteristic of the form $g(V) e^{j\alpha(V)}$ where V is the input envelope.[17] Then, if the input carrier is

$$v_i = V \sin(\omega t + \theta) \tag{1}$$

the output carrier is

$$v_o = g(V) \sin(\omega t + \theta + \alpha(V)) \tag{2}$$

With a narrowband modulated carrier of frequency ω, (1) denotes the instantaneous envelope and phase which are functions of time, and (2) is assumed to remain valid at each instant of time. Thus, the input/output characteristic is assumed to be frequency independent.

For convenience in the following, define $g(-V) = -g(V)$, so that a negative envelope is equivalent to a phase shift by π.

B.1.1 COMPUTATION OF 3RD-ORDER INTERMODULATION FOR TWO EQUAL SIGNALS

With two equal carriers at frequencies ω_1 and ω_2 accessing the channel simultaneously, the predominant intermodulation products are third-order at frequencies $2\omega_1 - \omega_2$ and $2\omega_2 - \omega_1$. The amplitude of the carriers in the output and of the third-order intermod products can be computed by a Fourier analysis.

The input consisting of two equal carriers is

$$v_i = \sin(\omega_1 t + \theta_1) + \sin(\omega_2 t + \theta_2)$$

$$= 2 \cos\left[\frac{(\omega_2 - \omega_1)t + \theta_2 - \theta_1}{2}\right] \sin\left[\frac{(\omega_2 + \omega_1)t + \theta_2 + \theta_1}{2}\right] \tag{3}$$

using a trignometric identity to write the input as a modulated carrier. Moving the time origin for convenience, (3) becomes

$$v_i = 2 \cos(\Delta\omega t/2) \sin(\omega_1 t + \Delta\omega t/2 + \theta) \tag{4}$$

where $\Delta\omega = \omega_2 - \omega_1$ and is assumed to be small compared with ω_1. The instantaneous envelope is

$$V = \left|2 \cos(\Delta\omega t/2)\right| \tag{5}$$

Using (2), and the definition of $g(-V)$, the output is*

$$v_o = g(2 \cos(\Delta\omega t/2))\sin(\omega_1 t + \Delta\omega t/2 + \theta + \alpha(|2 \cos(\Delta\omega t/2|))$$

$$= g(2 \cos(\Delta\omega t/2))\cos[\alpha(2|\cos(\Delta\omega t/2)|)] \sin(\omega_1 t + \Delta\omega t/2 + \theta)$$

$$+ g(2 \cos(\Delta\omega t/2))\sin[\alpha(2|\cos(\Delta\omega t/2)|)] \cos(\omega_1 t + \Delta\omega t/2 + \theta) \tag{6}$$

---

*Note, when $\cos(\Delta\omega t/2)$ is negative, there is a phase reversal by 180° on both the input carrier and the output carrier. The definition $g(-V) = -g(V)$ takes care of this phase reversal.

expressed in terms of quadrature components. Observe that the envelope functions
in (6) are periodic in $\Delta\omega/2$, are even functions of t, and have the same magnitude but
a reversed sign when $\Delta\omega t/2$ increases by π. Then, a Fourier expansion contains only
cosine terms and odd harmonics, and we write (with a factor of 2 for later convenience)

$$g(2\cos(\Delta\omega t/2))\cos[\alpha(2|\cos(\Delta\omega t/2)|)] = 2a_1\cos(\Delta\omega t/2) + 2a_3\cos(3\Delta\omega t/2) + \ldots$$

$$g(2\cos(\Delta\omega t/2))\sin[\alpha(2|\cos(\Delta\omega t/2)|)] = 2b_1\cos(\Delta\omega t/2) + 2b_3\cos(3\Delta\omega t/2) + \ldots \quad (7)$$

Substituting (7) into (6),

$$
\begin{aligned}
v_o = {} & a_1\sin(\omega_1 t + \theta) + a_1\sin(\omega_1 t + \Delta\omega t + \theta) \\[4pt]
& + b_1\cos(\omega_1 t + \theta) + b_1\cos(\omega_1 t + \Delta\omega t + \theta) \\[4pt]
& + a_3\sin(\omega_1 t - \Delta\omega t + \theta) + a_3\sin(\omega_1 t + 2\Delta\omega t + \theta) \\[4pt]
& + b_3\cos(\omega_1 t - \Delta\omega t + \theta) + b_3\cos(\omega_1 t + 2\Delta\omega t + \theta) \\[4pt]
& + \ldots\ldots \quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad (8)
\end{aligned}
$$

Thus,

$$\text{Desired signal amplitude} = (a_1^2 + b_1^2)^{0.5}$$

$$\text{3rd-order intermod} = (a_3^2 + b_3^2)^{0.5} \quad (9)$$

and these can be computed when $g(V)$ and $\alpha(V)$ are specified.

B.1.2   INTERMODULATION RESULTS FOR CHANNEL EXHIBITING LINEAR AM/PM CHARACTERISTIC

Figure B-1 shows a presumed phase shift function which approximates the
behavior of a TWT currently under development. Note that the phase shift does not
tend to flatten out near saturation. The AM/PM conversion coefficient, degrees/dB,
is constant as power is dropped until a maximum phase shift is reached. * Figures

*When the conversion coefficient in degrees/dB is constant, the phase shift in (2) is
proportional to the logarithm of the instantaneous envelope; i.e., $\alpha(V) = k\log(V)$.
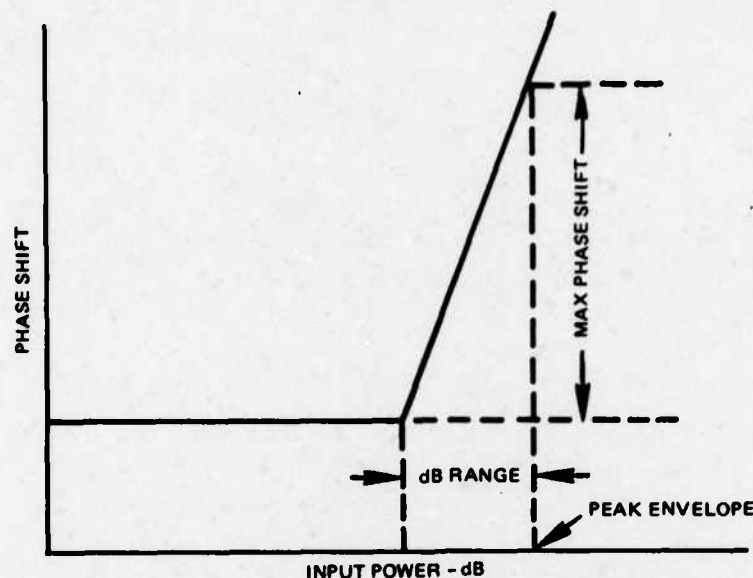
Figure B-1. Approximation to Phase Shift Behavior

B-2 and B-3 present computed results for 3rd-order intermod as a function of the conversion coefficient and the range of variation of phase shift, relative to the phase shift at the peak instantaneous input envelope. (The same data points are plotted with two different abscissas.)

From the results, it appears that AM/PM conversion coefficients below 3°/dB should not cause excessive intermod with QPSK/FDMA operation (taking 20 dB as a rough design goal). For coefficients greater than 3°/dB, the significant parameter appears to be the maximum phase shift produced as the instantaneous input envelope varies. A maximum phase shift in excess of about 15° could cause excessive intermod with QPSK/FDMA.

B.1.3 COMPUTED TWO-CARRIER INTERMODULATION FOR HUGHES' 40W TWT

Figure B-4 plots the single-carrier saturation curve and the phase shift curve for a 40W Hughes' TWT. Of course, the saturation curve of the tube should droop above the saturation point. Since measured data above saturation was not available, it was assumed that a limiter would prevent the instantaneous input envelope from exceeding +2 dB, at which point the phase shift is 66°. Thus, the saturation curve of the limiter in cascade with the TWT is taken as flat above 0 dB. The phase shift is assumed to be constant at 66° for inputs exceeding +2 dB. The AM/PM conversion

B-4
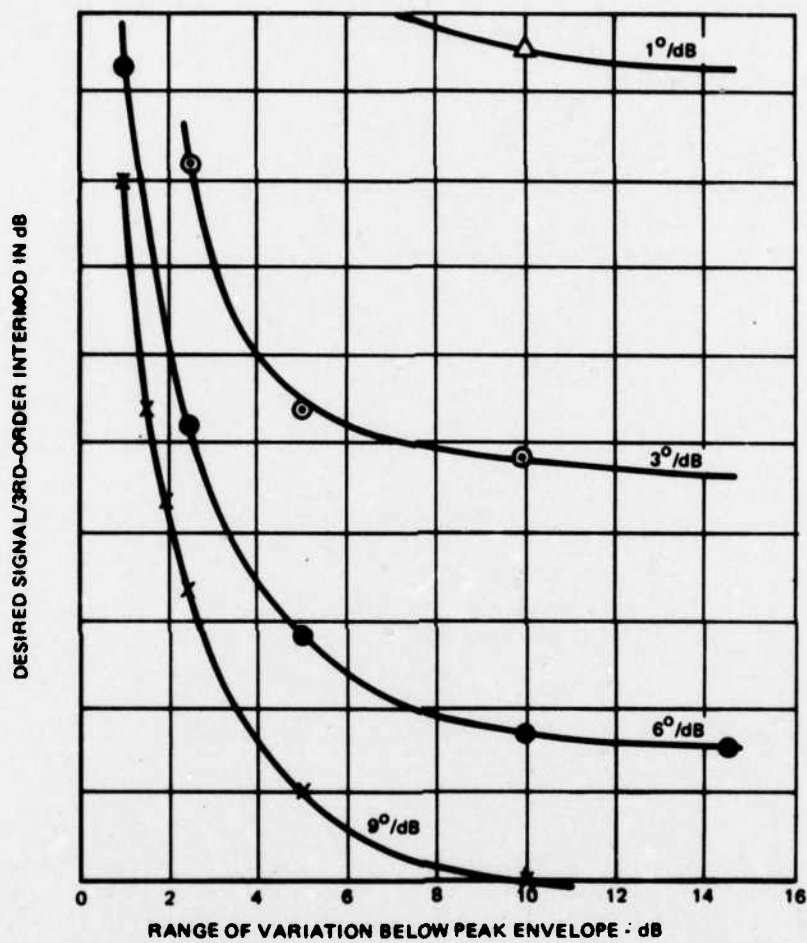
Figure B-2. 3rd-Order Intermod Due to AM/PM Conversion
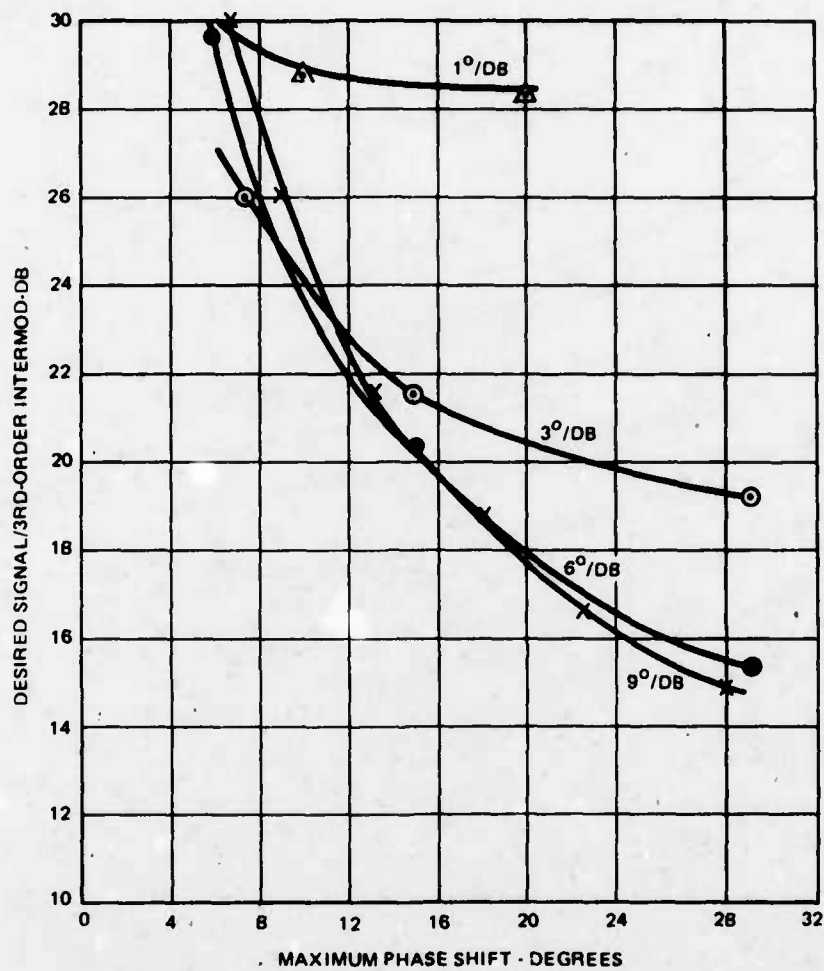
Figure B-3. 3rd-Order Intermod Due to AM/PM Conversion

776-3824
UNCLASSIFIED

B-6
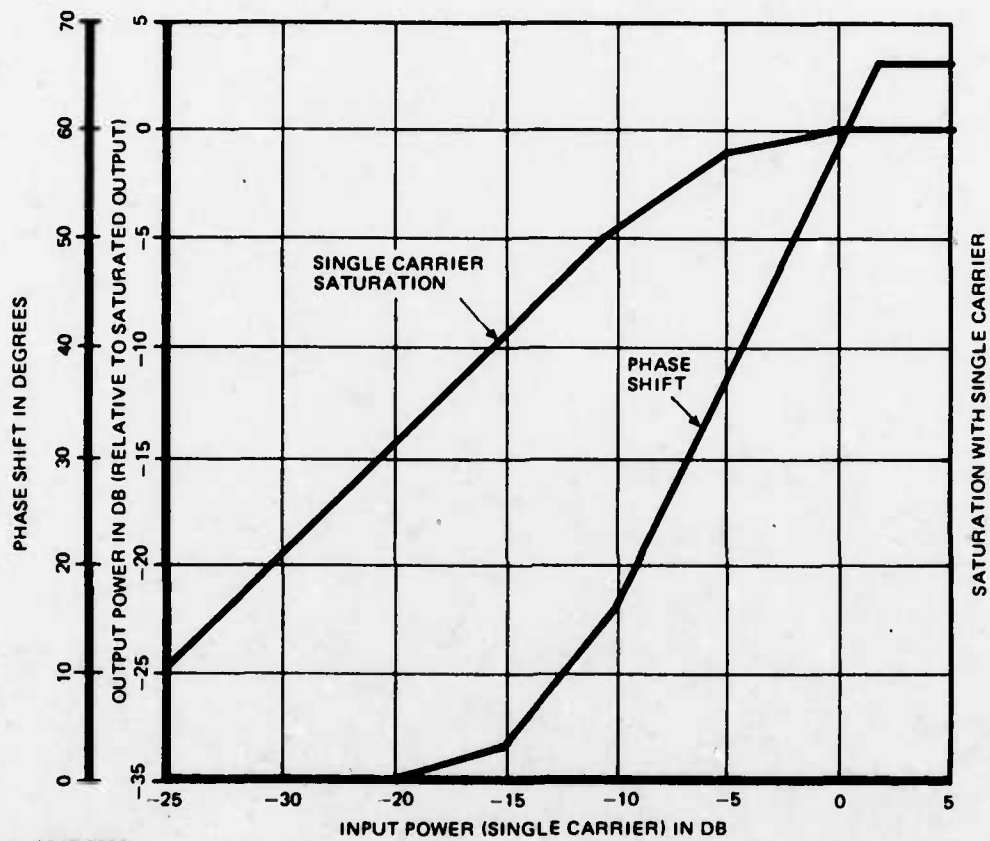
Figure B-4. Amplitude and Phase Characteristic of the Hughes' TWT

for the TWT is 4°/dB in the range -10 dB to 0 dB of input.

Figure B-5 plots the computed results. As mentioned, there is no droop in the output of a desired carrier because of the assumed limiter preceding the TWT.

Three third-order intermod curves are plotted in Figure B-5, all for the same saturation curve in Figure B-4. The solid curve in Figure B-5 is for the phase shift curve of Figure B-4. In addition, the cases of zero phase shift (saturation only) and additional phase shift are plotted. The latter scales the actual phase shift curve by 1.375, giving 5.5°/dB of AM/PM conversion and a maximum phase shift of 91°.

The significant increase in third-order intermodulation due to the phase shift may be observed. The phase shift has essentially no effect on the output at the desired frequency, and only one curve is, therefore, plotted in Figure B-5.

B.1.4   INTERMODULATION DISTORTION WITH THREE EQUAL SIGNALS

We now consider the case of three carriers simultaneously accessing the channel. In contrast to the two-carrier case, the intermodulation products can fall upon the desired outputs when the carriers are equally spaced in frequency. The effect is to produce distortion of amplitude and phase on the desired output.

For three equal amplitude and equally spaced carriers, the input is

$$v_i = \cos(\omega t - \Delta\omega t - \theta_1) + \cos(\omega t) + \cos(\omega t + \Delta\omega t + \theta_2)$$

$$= \text{Re}\left\{(I + jQ)e^{j\omega t}\right\} \tag{10}$$

where θ_1 and θ_2 express the relative phasing of the carriers. Usually, θ_1 and θ_2 are uniformly distributed since the carriers are from independent transmitters. In (10), the quadrature components

$$I(t) = 1 + \cos(\Delta\omega t + \theta_1) + \cos(\Delta\omega t + \theta_2)$$

$$= 1 + 2\cos(\Delta\omega t + \theta)\cos\phi$$

$$Q(t) = \sin(\Delta\omega t + \theta_2) - \sin(\Delta\omega t + \theta_1)$$

$$= 2\cos(\Delta\omega t + \theta)\sin\phi \tag{11}$$
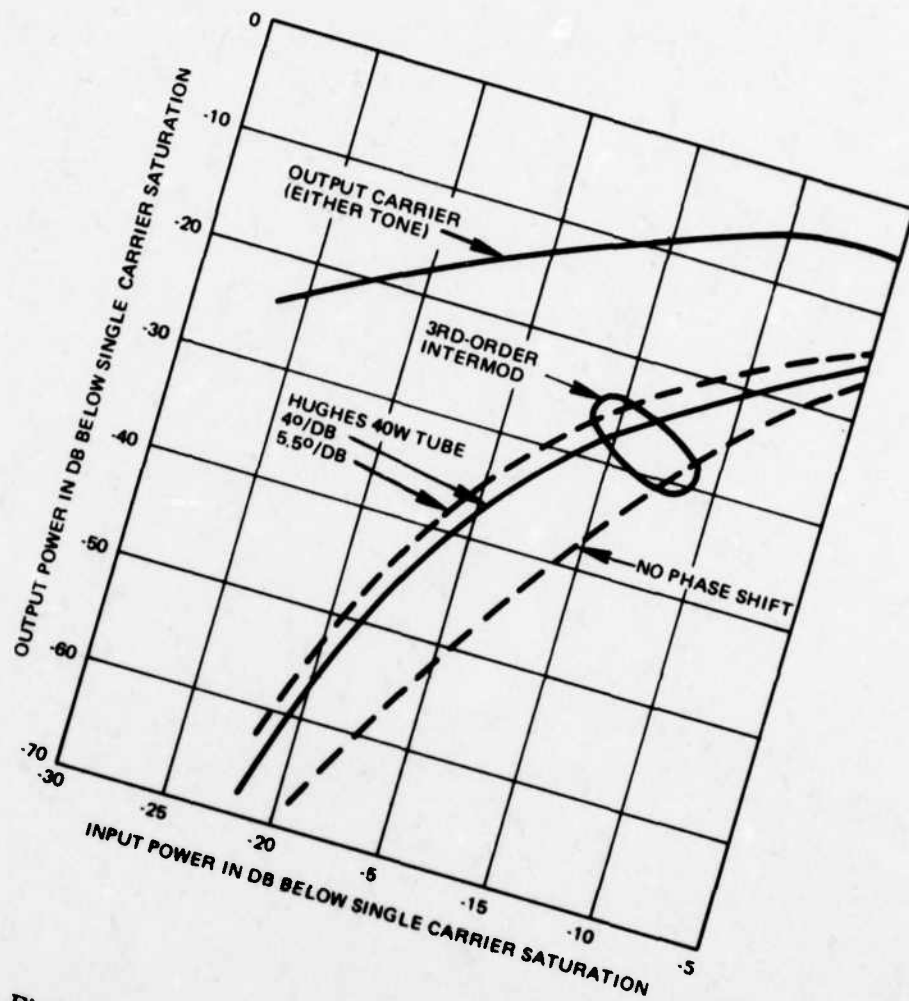
Figure B-5.   Two Equal Carriers in TWT

are defined, and we introduce the phasing parameters $\theta = (\theta_1 + \theta_2)/2$ and $\omega = (\theta_2 - \theta_1)/2$. It is convenient to retain, henceforth, only the complex modulation and suppress $e^{j\omega t}$. Then, the (complex) output modulation of the channel is

$$v_o = g(V)(I + jQ)e^{j\alpha(V)} \tag{12}$$

where

$$V = \sqrt{I^2 + Q^2} = [1 + 4\cos(\Delta\omega t + \theta)\cos\omega + 4\cos^2(\Delta\omega t + \theta)]^{.5} \tag{13}$$

is the instantaneous envelope entering into the phase shift and saturation functions.

For a fixed θ and ω, the output modulation is a periodic function of time with fundamental frequency $\Delta\omega$, and the harmonic content can be obtained by a Fourier analysis. We are interested in the frequency component at zero modulation frequency, corresponding to the output at the center carrier. This component is

$$v_o]_0 = \frac{1}{2\pi} \int_0^{2\pi} g(V)(I + jQ)e^{j\alpha(V)} \, d(\Delta\omega t) \tag{14}$$

which is a function of ω, but by inspection of (11) and (13) is independent of θ. For randomly phased input carriers, ω is uniformly distributed over 2π.

In the above, we have derived the intermodulation effect on the center carrier of the three equally spaced input carriers. The effect is to produce a phase shift and an amplitude change dependent on the relative phasing of the input carriers. A similar analysis was previously carried through considering just amplitude saturation[18]. Given $g(V)$ and $\alpha(V)$, for the channel, (14) can be numerically integrated, and is a function of the relative phasing parameter ω.

In a FDMA application, the above analysis shows how intermodulation degrades the operation with multiple carriers in the channel. For instance, with BPSK or QPSK data transmission, the phase shift and amplitude reduction due to intermodulation reduces the noise margin and thereby increases the required E_b/N_o. We can see this by computing a numerical example with a postulated phase shift function.

If the channel has a constant AM/PM coefficient $k_{AM/PM}$ in degrees/dB, the phase shift function in degrees is

$$\alpha(V) = 20k_{AM/PM} \log_{10}(V) \tag{15}$$

More generally, the phase shift function can be assumed to be constant until V exceeds a value V_o, in which case

$$\alpha(V) = 0 \qquad\qquad ; \ V < V_o$$

$$= 20 k_{AM/PM} \log_{10}(V/V_0) ; \ V > V_o \qquad\qquad (16)$$

Now, the effect of the AM/PM conversion will depend on the input average power level relative to V_o^2.

The integral in (14) has been numerically computed by approximating it as a summation over 128 points. The result is a complex quantity obtained for 32 equally spaced values of φ and plotted to show the variation of output envelope and phase for the center carrier. Figures B-6 to 10 give the results. Note that 0 dB is defined as the power level where the linear phase shift with dB begins, as defined by V_o^2 from (16).

Figure B-6 has $k_{AM/PM} = 2^o/dB$, and the average input power level is 6 dB above V_o^2. The maximum angular spread of the output phase of the center carrier is seen to be about 11°. (This is the marked angle corrected for the different horizontal and vertical scales in the plot.) In Figures B-7 to B-10, $k_{AM/PM} = 6^o/dB$. Note that if the input power level equals V_o^2, the angular spread is 18°, and is considerably less for lower input levels. However, if the input power level exceeds V_o^2 by 3 dB or more, the angular spread exceeds 30°. Also, the output envelope of the center carrier varies by almost 5 dB.

Based on the above results, it may be concluded that an AM/PM conversion coefficient of $2^o/dB$ will not seriously impair QPSK operation in FDMA, since the noise margin of 45° is eroded by only 5.5^o. On the other hand, if the coefficient is $6^o/dB$, the 45° noise margin can be eroded by 15° or more, and there is a substantial E_b/N_o penalty for QPSK operation.
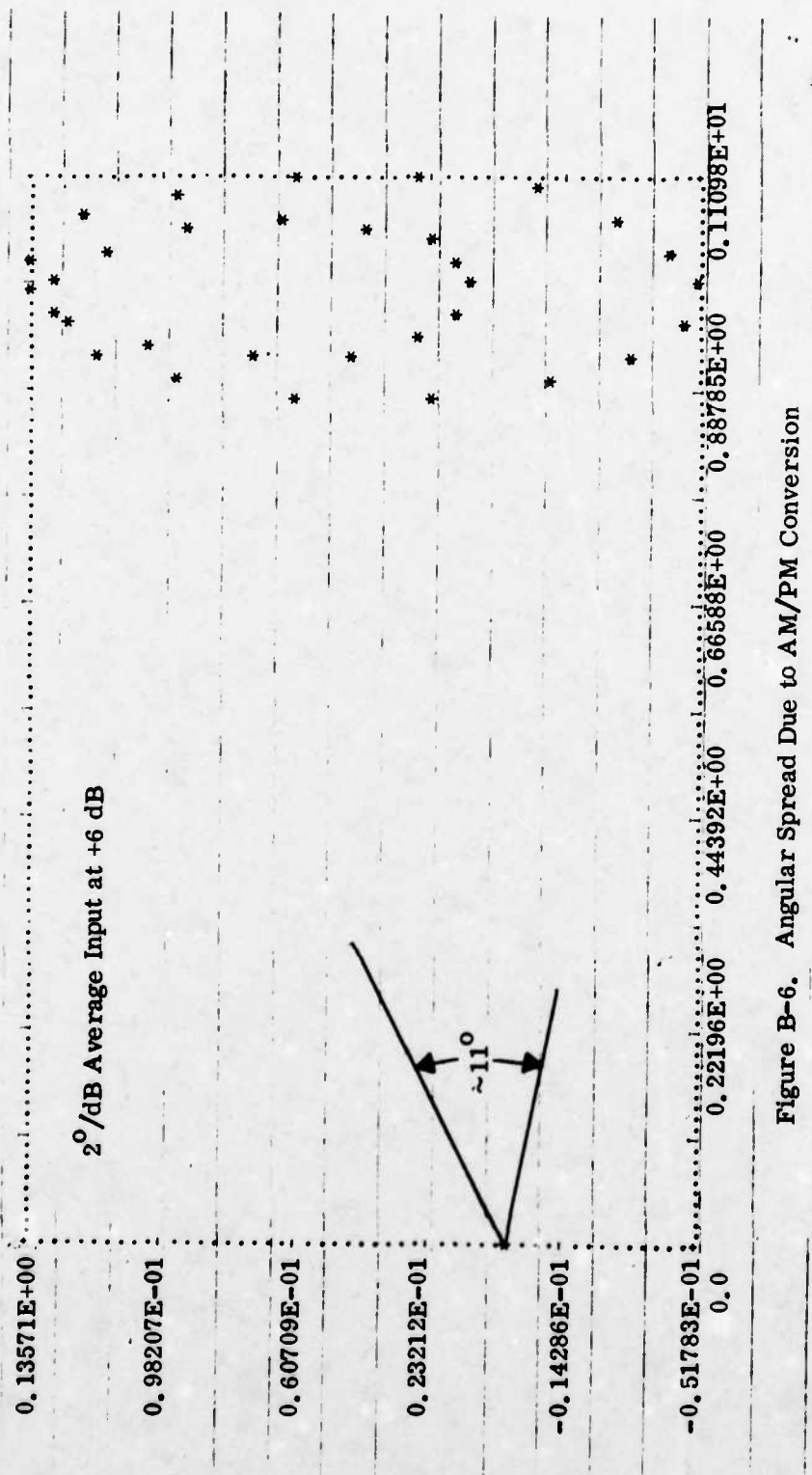
Figure B-6. Angular Spread Due to AM/PM Conversion

2°/dB Average Input at +6 dB

~11°

0.13571E+00
0.98207E-01
0.60709E-01
0.23212E-01
-0.14286E-01
-0.51783E-01
0.0

0.22196E+00   0.44392E+00   0.66588E+00   0.88785E+00   0.11098E+01
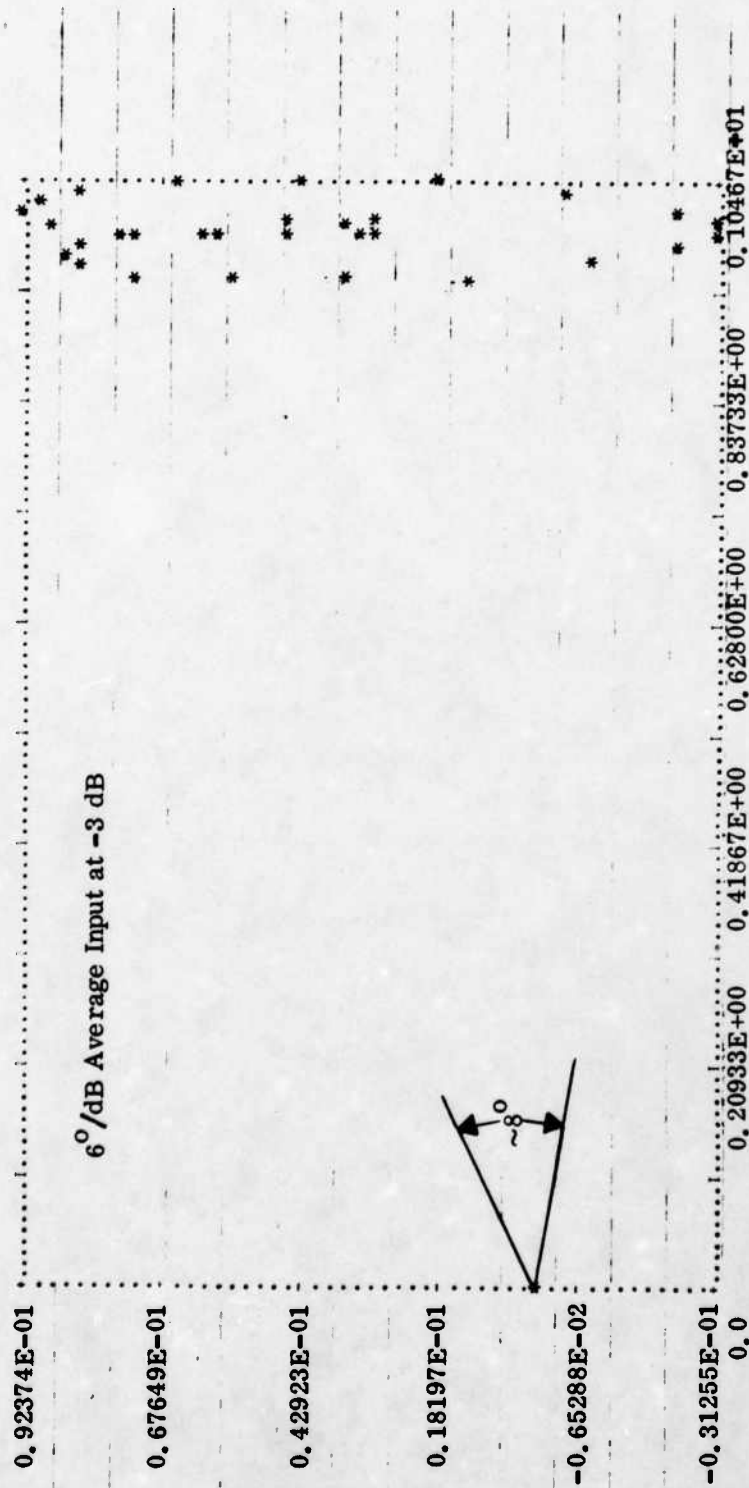
Figure B-7. Angular Spread Due to AM/PM Conversion

6°/dB Average Input at -3 dB

0.92374E-01

0.67649E-01

0.42923E-01

0.18197E-01

-0.65288E-02

-0.31255E-01

0.0   0.20933E+00   0.41867E+00   0.62800E+00   0.83733E+00   0.10467E+01
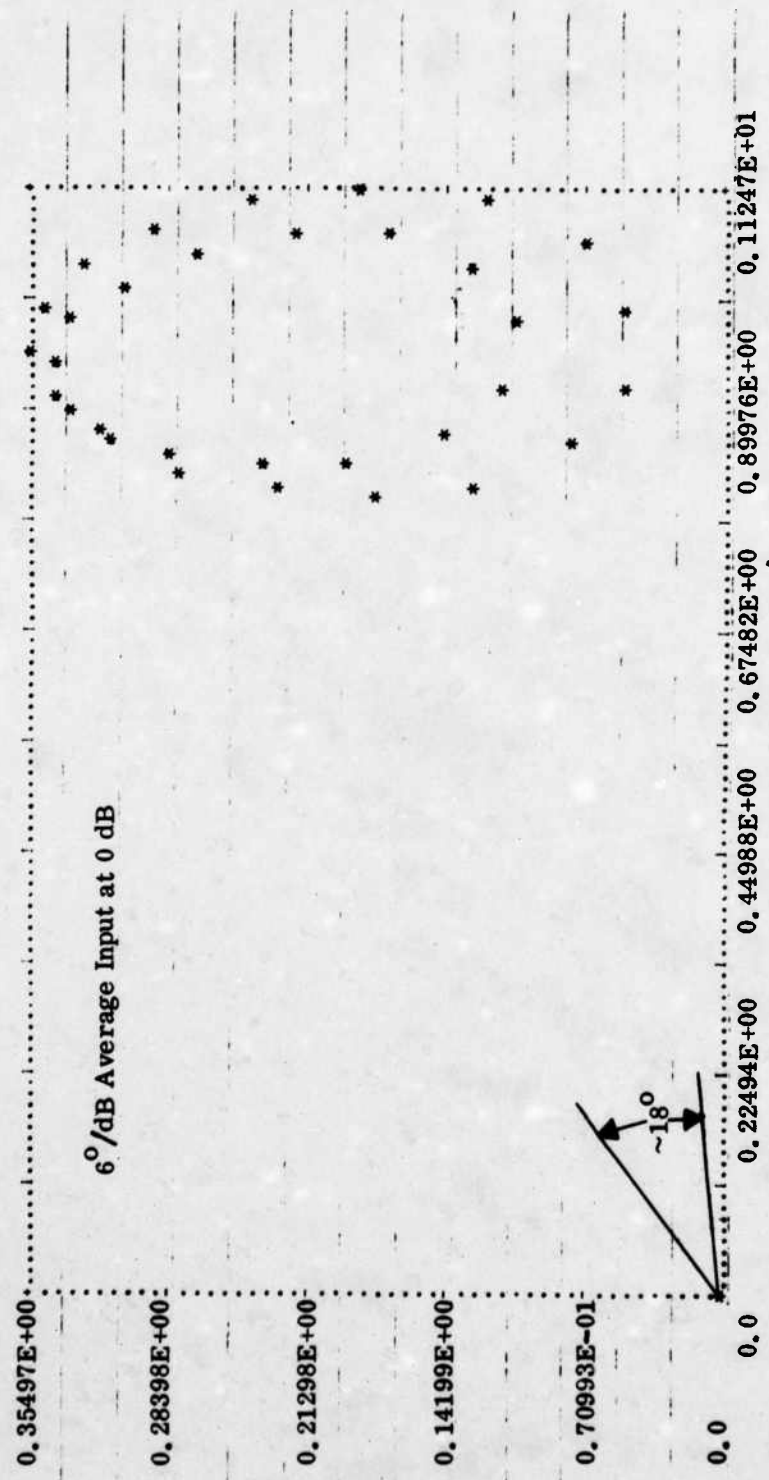
Figure B-8. Angular Spread Due to AM/PM Conversion

B-14

6°/dB Average Input at +3 dB

~30°

0.36550E+00

0.27951E+00

0.19352E+00

0.10753E+00

0.21544E-01

-0.64443E-01

0.0

0.24149E+00 0.48297E+00 0.72446E+00 0.96594E+00 0.12074E+01
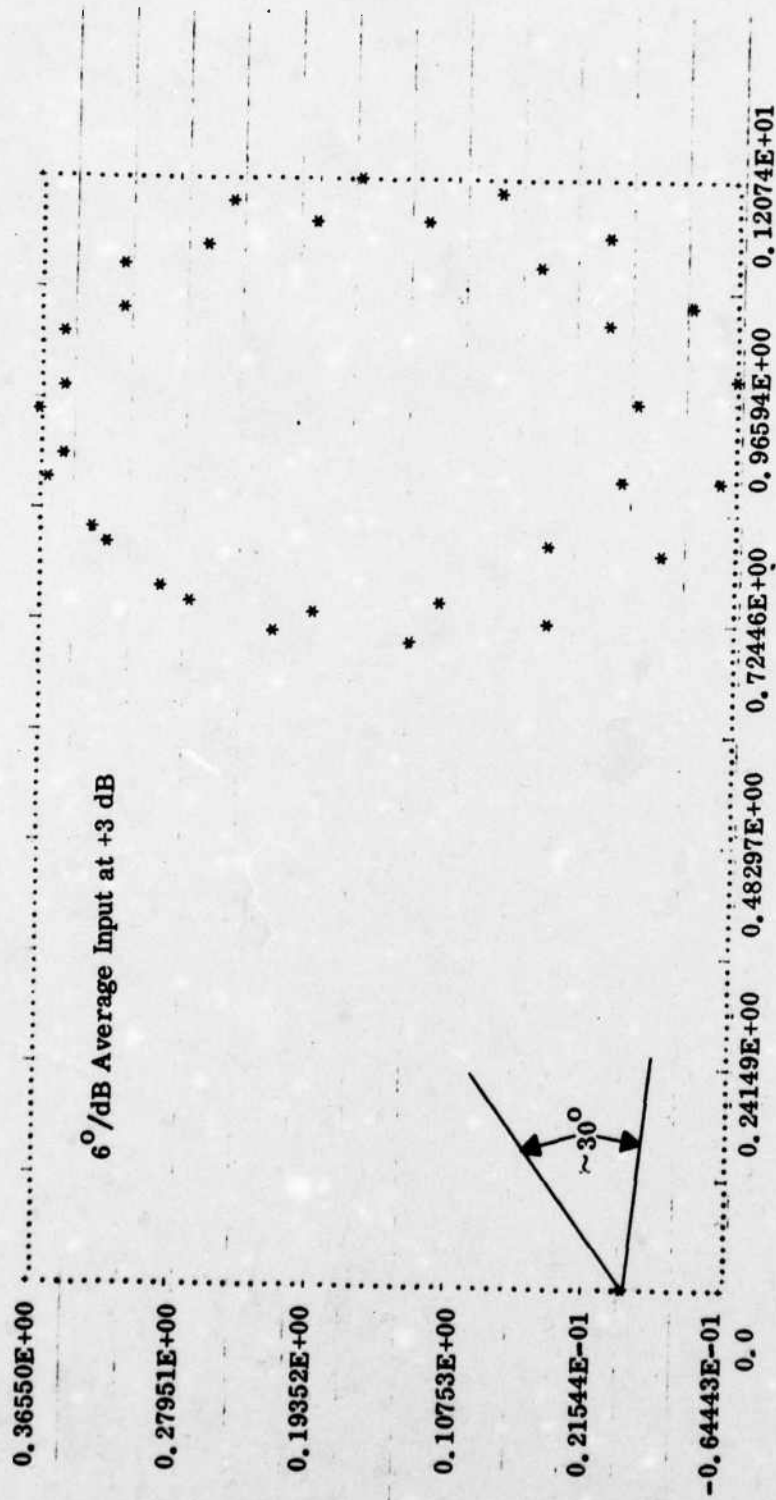
Figure B-9. Angular Spread Due to AM/PM Conversion

6°/dB Average Input at +6 dB

~33°

0.39188E+00

0.29559E+00

0.19930E+00

0.10301E+00

0.67235E-02

-0.89565E-01

0.0

0.24621E+00   0.49243E+00   0.73864E+00   0.98486E+00   0.12311E+01
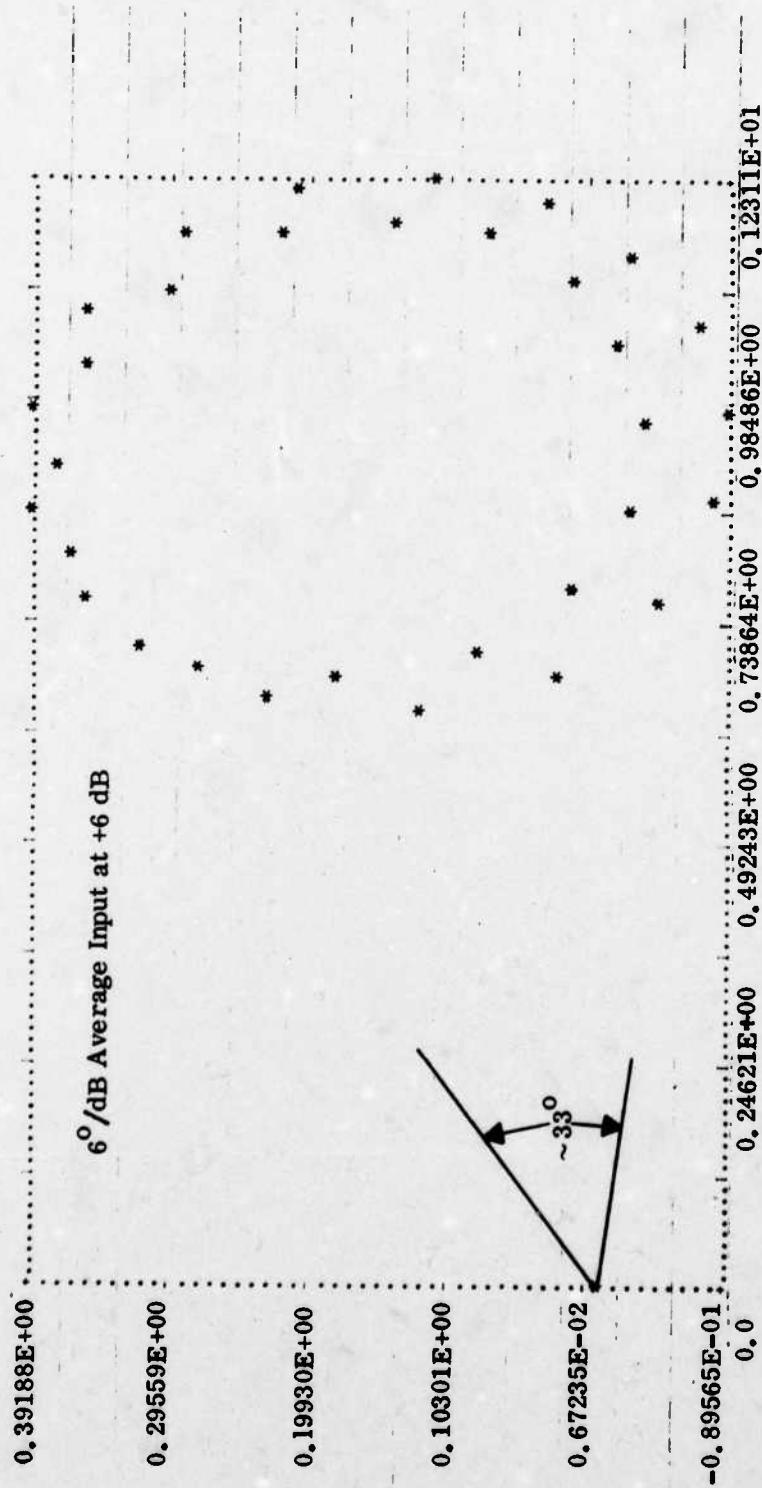
Figure B-10. Angular Spread Due to AM/PM Conversion

B-16

B.2 SIGNAL-TO-INTERMODULATION NOISE FOR NONLINEAR AMPLIFIERS

The case of a large number of signals accessing the channel can be approximated by representing the composite input signal as a Gaussian noise process whose power is equal to that of the combined signals to be passed through the non-linearity. The signal-to-intermodulation ratio (SIR) at the output of the nonlinearity is an important measure of the degradation produced.

This report covers a technique for modelling complex nonlinear transfer functions, based on reference 19, and a method for separating the output spectrum into the signal component and the intermodulation component. A computer program has been developed to model nonlinear transfer functions, generate input signal processes, and compute the output spectra and SIR. Results of the simulation are presented for the cases of a hard-limiter, a linear AM-PM conversion characteristic, and a Hughes' TWT characteristic.

B.2.1 TECHNICAL ANALYSIS

Only the first-zone spectral components of the nonlinear amplifier output are considered in this analysis. This component $z(t)$ represents an output process whose bandwidth is less than an octave surrounding the center frequency ω_o, assuming the narrow-band input signal

$$x(t) = \sqrt{2}\,\sigma(t)\,\cos[\omega_o t + \theta(t)] \tag{17}$$

Since $x(t)$ is narrow-band, the nonlinearity is assumed to be independent of frequency. The complex transfer function of the nonlinear device contains an output envelope voltage and phase shift which are dependent on the rms instantaneous input envelope voltage $\sigma(t)$. The output can be expressed as

$$z(t) = \sqrt{2}\,G[\sigma(t)]\,\cos\{\omega_o t + \theta(t) + \omega[\sigma(t)]\} \tag{18}$$

$G[\sigma]$ represents the complex envelope amplitude-distorting transfer function and $\omega[\sigma]$ represents the envelope-dependent phase shift.

Following [19], a nonlinear device with arbitrary amplitude and phase shift characteristics is modelled as the parallel combination of two memoryless nonlinearities whose inputs have statistics identical to those of the original input signal and whose outputs are linearly independent for the narrow-band case considered. This independence results in the capability of adding the power spectra and separates the in-phase and quadrature components of the nonlinearity.

The quadrature version of equation (2) becomes

$$z(t) = \sqrt{2}\, G_p[\sigma(t)]\, \cos\{\omega_o t + \theta(t)\} - \sqrt{2}\, G_q[\sigma(t)]\, \sin\{\omega_o t + \theta(t)\} \qquad (19)$$

This form suggests the model shown in Figure B-11 where the in-phase and quadrature nonlinearities are simple amplitude-distorting nonlinearities with transfer functions $G_p[\sigma]$ and $G_q[\sigma]$, respectively, where

$$G_p[\sigma] = G[\sigma]\, \cos \varphi[\sigma] \qquad (20)$$

$$G_q[\sigma] = G[\sigma]\, \sin \varphi[\sigma] \qquad (21)$$

Due to the narrow-band assumption, the 90° phase shifter in Figure B-11 has no effect on the statistics of the signal $x(t)$ as it passes through. Since both of the nonlinearities $G_p[\sigma]$ and $G_q[\sigma]$ are now only amplitude nonlinearities which contain no phase distortion, the inverse first-order Chebyshev transform can be used to get the equivalent instantaneous voltage transfer functions $g_1(x)$ and $g_q(x)$ as in reference 20. Since the in-phase and quadrature amplitude nonlinearities do not affect the phase of the signals passing through them, their respective outputs are the quadrature components of a new narrow-band random process and are linearly independent. Due to this independence, the two individual power spectra add to give the spectrum of the total output.
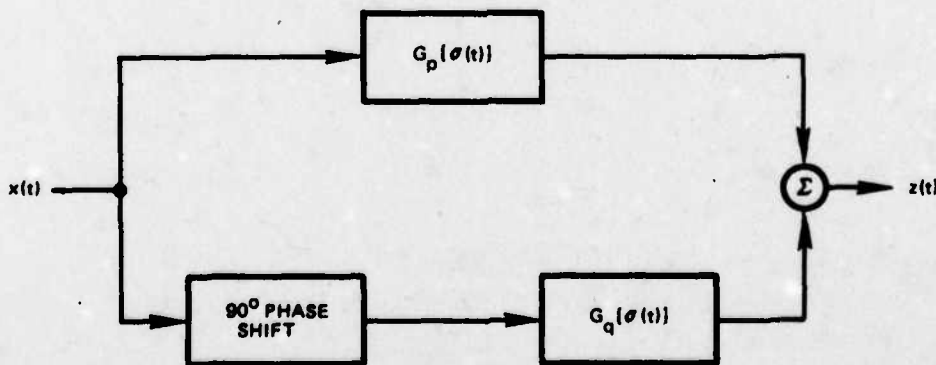
776-3826
UNCLASSIFIED

Figure B-11. Quadrature Model of Bandpass Nonlinearity

B-18

B.2.2   COMPUTATION OF POWER SPECTRA

A Fourier series is used to compute the power spectra of the outputs of the two amplitude nonlinearities. The instantaneous voltage function becomes

$$g(x) = \sum_{\ell=1}^{L} b_\ell \sin\frac{(2\ell - 1)\pi x}{R} , \quad -\frac{R}{2} \leq x \leq \frac{R}{2} \tag{22}$$

As given, the period of this Fourier series is 2R and the representation must be valid over a range of width R about zero as illustrated in Figure B-12.

The coefficients, b_ℓ, of the Fourier series can be obtained by substituting $x(t) = \sqrt{2} \sigma \cos(\omega_0 t + \theta)$ in series (22) or by use of Blachman's transforms T2 and T7. The envelope transfer function corresponding to (22) thereby becomes

$$G[\sigma] = \sqrt{2} \sum_{\ell=1}^{L} b_\ell J_1 \left[\frac{\sqrt{2} \sigma (2\ell - 1)\pi}{R} \right] , \quad \sigma \leq \frac{R}{2\sqrt{2}} \tag{23}$$

where $J_1(\cdot)$ is the Bessel function of the first kind of order one.

The autocorrelation of the output of the nonlinear device is obtained as the next step. A model often employed to determine the quality of a link is the noise power ratio as measured by notching out a narrow band of the input signal spectrum
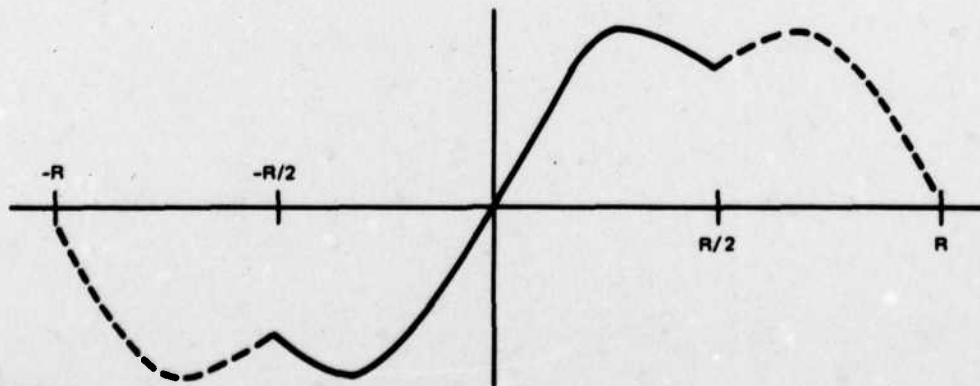
776-3827
UNCLASSIFIED

Figure B-12.   Representation of an Instantaneous Voltage Transfer Function by a Fourier Series

and measuring the amount of intermodulation energy falling into this band. To approximate the case of many channels through the nonlinearity, a similar procedure will be used. Shaped noise will be input to the nonlinear amplifier to represent the multiple channels and then the first-order noise and the sum of all other higher order noise harmonics at the bandpass filter output will be calculated. The ratio of these two represents the signal-to-interference ratio (SIR). References 21 and 22 follow this approach. In this representation, the output autocorrelation function $R_z(\tau)$ is given by a power series of the baseband equivalent input autocorrelation function, $R_0(\tau)$.

Rewrite equation (22) as

$$g(x) = j \sum_{\substack{k=-2L+1 \\ k \text{ odd}}}^{2L-1} c_k e^{j \, kax} \tag{24}$$

by letting $k = 2\ell - 1$, and $a = \pi/R$. Note that

$$c_{-k} = -c_k \tag{25}$$

The problem of computing the output of nonlinear devices for noisy input is considered in [21]. In this case, the output autocorrelation function, $R_z(\tau)$, is given by a power series of the input autocorrelation function, $R(\tau)$.

$$R_z(\tau) = \sum_{k=1}^{\infty} \frac{h_{ok}^2}{k!} R^k(\tau) \tag{26}$$

where

$$h_{ok} = \frac{1}{2\pi j} \int_c f(s) s^k e^{R(0)s^2/2} \, ds \tag{27}$$

$R(0)$ represents the input noise power. $f(s)$ is the bilateral Laplace transform of the nonlinear transfer function and the contour c is along the imaginary axis of the complex s plane. In terms of equation (24), equation (27) can be rewritten as

$$h_{ok} = \sum_{\ell \text{ odd}} c_\ell \, (j\ell a)^k \, e^{-R(0)\ell^2 a^2/2} \tag{28}$$

For the bandpass case

$$R(\tau) = R_0(\tau) \cos \omega_0 \tau \tag{29}$$

where $R_0(\tau)$ represents the baseband input process. Using (28) and recognizing that the summation over k in (26) can be expressed as a series expansion of exponentials, the output correlation function, $R_z(\tau)$, can be written

$$R_z(\tau) = -\sum_{\ell \text{ odd}} \sum_{m \text{ odd}} c_\ell c_m e^{-R(0) a^2 (\ell^2 + m^2)/2} e^{-R_0(\tau) a^2 \ell m} \cos \omega_0 \tau \qquad (30)$$

To obtain the baseband equivalent, $R_y(\tau)$, from the total output autocorrelation function, $R_z(\tau)$, we will first expand $R_z(\tau)$ in a Fourier series and select the component in the first zone (surrounding ω_0). Therefore, assuming ω_0 large, the Fourier coefficient of the first zone component is given by the Fourier integral

$$\frac{\omega_0}{\pi} \int_0^{2\pi/\omega_0} R_z(\tau) \cos \omega_0 \tau \, d\tau = \frac{1}{\pi} \int_0^{2\pi} e^{-R_0(\tau) a^2 m \cos \theta} \cos \theta \, d\theta$$

$$= -2I_1(R_0(\tau) a^2 \ell m) \qquad (31)$$

where $I_1(b)$ is the modified Bessel function of order one. Thus, the expression for the baseband equivalent of the first zone output, $R_y(\tau)$, becomes

$$R_y(\tau) = 2 \sum_{\ell \text{ odd}} \sum_{m \text{ odd}} c_\ell c_m e^{-R(0) a^2 (\ell^2 + m^2)/2} I_1(R_0(\tau) a^2 \ell m) \qquad (32)$$

Equation (32) gives the autocorrelation function of the baseband equivalent of the first zone output which includes the undistorted component $R_{y_0}(\tau)$ and the harmonic components, $\tilde{R}_y(\tau)$, which fall in the first zone. As shown in reference 23, when the input signal is a Gaussian random process, the undistorted component $R_{y_0}(\tau)$ is given by the term for k = 1 in the summation in equation (26); i.e., $R_{y_0}(\tau)$ is a linear function of $R_0(\tau)$. $R_{y_0}(\tau)$ can therefore be obtained from equations (26) and (28) by identifying the component proportional to $R_0(\tau)$. The factor multiplying $R_0(\tau)$ represents a gain factor through the nonlinear device

$$R_{y_0}(\tau) = \left[\sum_{\ell \text{ odd}} a\ell \, c_\ell \, e^{-R(0)a^2\ell^2/2} \right]^2 R_0(\tau) \qquad (33)$$

The intermodulation autocorrelation function $\tilde{R}_y(\tau)$ is found by subtracting equation (33) from equation (32). The various spectra are obtained by Fourier transforming these autocorrelations. The total signal-to-intermodulation power ratio (SIR) is given

by dividing $R_{y_0}(0)$ by $\widetilde{R}_y(0)$. The SIR within a given frequency band is given by the ratio of the spectra $S_{y_0}(\omega)$ and $\widetilde{S}_y(\omega)$ over the desired frequency range.

B.2.3   INPUT SPECTRUM

A Gaussian-shaped spectrum was assumed as input to the nonlinear amplifier. This represents simultaneous access by a large number of independent carriers. For such an assumed spectrum, the baseband autocorrelation is expressed as

$$R_0(\tau) = \sigma_0^2 e^{-(\beta\tau)^2/2} \tag{34}$$

where σ_0^2 represents the total power in the input process and β scales the spectrum in the frequency domain.

B.2.4   NONLINEAR AMPLIFIER MODELS USED

Three nonlinear characteristics were assumed. The first represents a hard limiter defined in terms of equation (18) as

$$G(\sigma) = 1$$
$$\wp(\sigma) = 0 \tag{35}$$

The second model represents a linear AM-PM conversion characteristic where the amplitude transfer characteristic is linear and the phase shift increases linearly as a function of the instantaneous input power in dB. Below the input drive power $P_F = 10$ dB, the phase shift is zero. This phase shift characteristic due to AM-PM conversion is given in Figure B-1. The slope of the characteristic is dependent on the number of degrees per dB change of input drive level.

The third characteristic is that of a Hughes' TWT. The amplitude and phase transfer characteristics are shown in Figure B-4, as a function of input drive. Characteristics of the TWT are not known for inputs greater than 0 dB, but TWT devices normally exhibit a relatively constant output for some range above 0 dB before the output begins to fall with increasing drive. Therefore, for input power above 0 dB, the amplitude is hard-limited to the 0 dB level.

B.2.5   COMPUTER PROGRAM NONLIN

The Fortran IV computer program NONLIN models the nonlinearities and performs all computations necessary to obtain the spectra and the signal-to-intermodulation power ratio SIR.

Initially, the computer program sets up the in-phase and quadrature components, $G_p(\sigma)$ and $G_q(\sigma)$, based on the nonlinearity being modelled. The nonlinearity is specified at 40 points over the range $\sigma \epsilon [0, 3]$. Next, equation (23) is solved to get the $(L = 14)$ coefficients $\{b\}$. This is accomplished by minimizing the rms error between the specified values, $G_p(\sigma)$ or $G_q(\sigma)$, at the 40 nodes and the values obtained at these nodes by the approximation (23) given a set of coefficients b. The set of linear equations are differentiated with respect to the coefficients and subroutine SIMQ is used to solve the resulting simultaneous equations. Using the resulting coefficients, the characteristic is generated at 80 points in the range $\sigma \epsilon [0, 3]$ to determine the accuracy of the fit obtained.

The Gaussian input baseband autocorrelation function $R_o(\tau)$ is next generated and the input power is stepped over the range $[-20, 0]$, in decibels. The autocorrelation is defined at 256 points.

Equations (32) and (33) are solved to determine $R_{y_o}(\tau)$ and $R_{y_o}(\tau)$. $\tilde{R}_y(\tau)$ is given by subtracting $R_{y_o}(\tau)$ from $R_y(\tau)$. These operations are performed separately for the in-phase and quadrature correlation functions.

The resulting autocorrelations $R_y(\tau)$ and $\tilde{R}_y(\tau)$ are then Fourier transformed to obtain the corresponding output spectra. Program NONLIN then displays the total SIR in the output and the SIR at the center (ω_o) of the output spectra.

B.2.6   RESULTS OF THE SIMULATION

Figure B-13 shows the results obtained for the three nonlinearities modelled. The signal-to-intermodulation ratio (SIR), in decibels, is the ratio measured at bandcenter ω_o. The abscissa gives the input power in dB, where 0 dB equals unit power.

The lowest SIR is produced by the hard limiter. For high drive levels approximately 9 dB of SIR is produced, in agreement with theory. For drive levels less than -10 dB, the SIR uses due to the inaccuracy of the Bessel series expansion for small input drive levels.
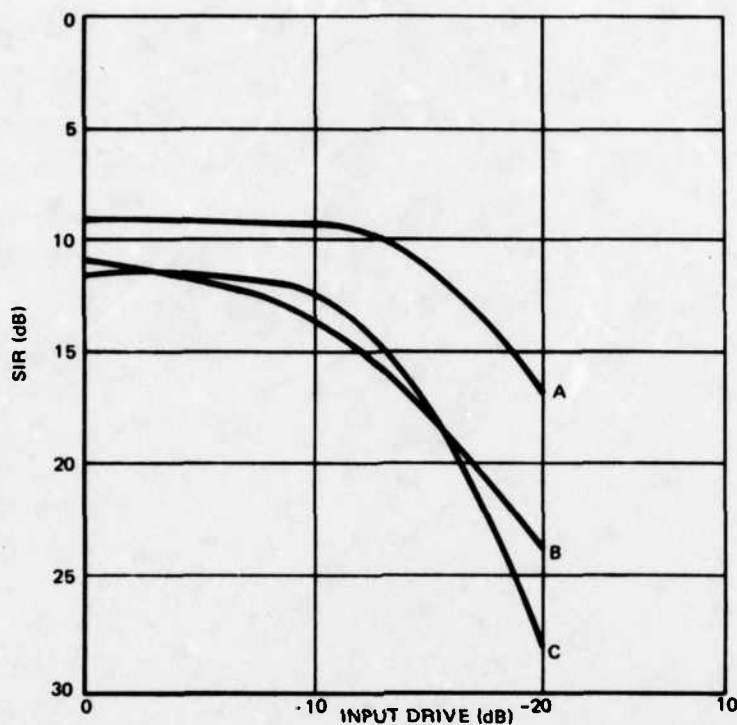
Figure B-13. Signal-to-Intermodulation Ratios for Three Nonlinear Devices
(A = hard limiter; B = Hughes' TWT; C = linear AM/PM (6°/dB))

The composite Hughes' TWT characteristic produces about 11 dB of SIR for 0 dB input and the SIR increases in a linear fashion for large backoff. Note that 0 dB of input drive corresponds to 0 dB of backoff for the TWT.

The linear AM-PM conversion characteristic has a SIR which increases rapidly for low drive levels and the SIR levels off at about 11.5 dB for drive levels in the range of -8 dB to 0 dB. The phase shift modelled was 6°/dB.

In conclusion, the results obtained by the analytical model used in this study are in good agreement with results obtained elsewhere, such as in Section 7. The technique permits modelling a variety of nonlinear device characteristics and input signal spectra.

APPENDIX C
ADAPTIVE EQUALIZATION CONTROL ALGORITHMS

Many communication systems used for the transmission of biphase or quadriphase-modulated digital data contain a variety of channel distortions which impair the overall performance of the system. One approach for minimizing this degradation involves eliminating or minimizing these errors by setting rigid specifications on allowable phase distortion. This method can only be carried to the point at which either cost or state-of-the-art limitations set bounds beyond which further minimization of phase error is not possible. In other instances the channel is already established and existing phase distortions cannot be altered.

A second approach involves phase equalization of the received signals to minimize existing system phase distortions. Such equalization can be of the fixed type if the overall phase distortion is known and time-invariant or adaptive if the distortion is unknown or time-varying or if the channel configuration changes. The following discussion concentrates on adaptive equalizers and algorithms to control the adaptation, based on either transmission of a known training sequence or a form of decision-directed feedback. The two control algorithms discussed both minimize the mean square error. The LMS algorithm is an iterative solution of the steepest-descent type[24], while the TRENCH algorithm[25] is a matrix-inversion solution.

C.1 ADAPTIVE EQUALIZER STRUCTURE

The adaptive equalizer assumes the form of a transversal filter (or non-recursive digital filter) as in Figure C-1, in which the tap weights are adaptively changed to minimize an error norm. Design of the adaptive equalizer involves choice of the error norm and selection of the technique for adaptation. The error norm is typically the mean square error between the equalized time response $s(t)$ and the desired response $d(t)$. In Figure C-2 the received samples in the equalizer at time t are $x_1(t), \ldots, x_n(t)$, assuming an n-tap equalizer. These samples are multiplied by the tap weights $w_1, \ldots, w_n$ as in Figure C-1 and summed to produce the output $s(t)$. $d(t)$ is the corresponding time sample of the desired signal.

DIRECTION OF SHIFT →

X (nk) → | X (1) | X (2) | X (3) | X (4) | .... | X (n-1) | X (n) |

w₁ w₂ w₃ w₄ .... w₍ₙ₋₁₎ wₙ

....

Σ

OUTPUT
s (t)
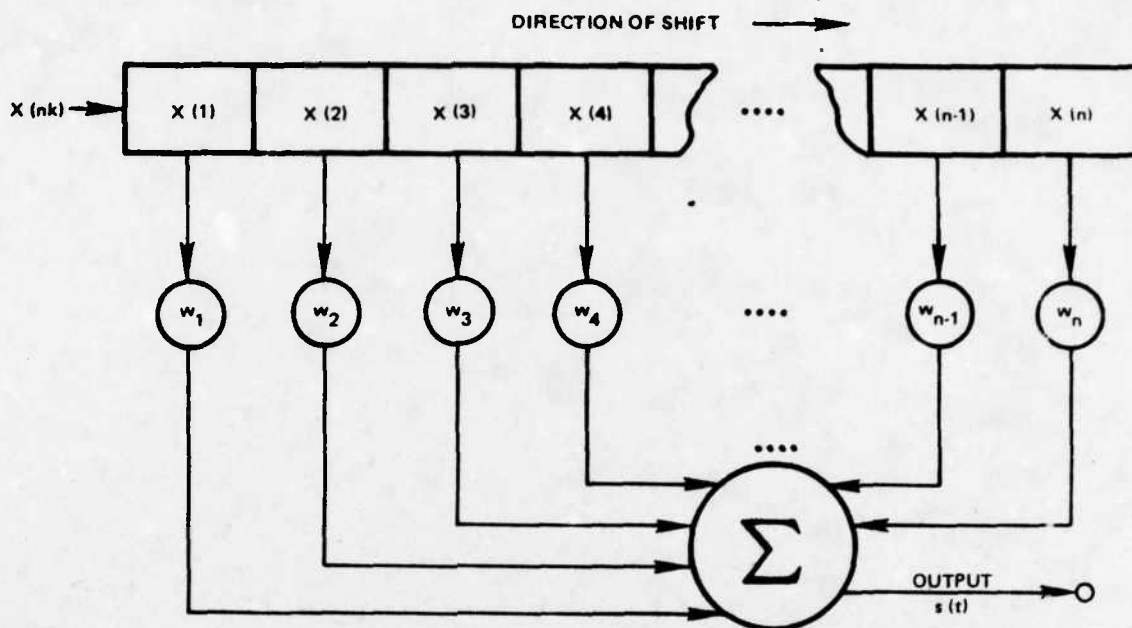
175-99
UNCLASSIFIED

Figure C-1. Transversal Filter

X (nk) → | X (1) | X (2) | .... | X (n-1) | X (n) |

w₁
w₂
⋮
w₍ₙ₋₁₎
wₙ

Σ → OUTPUT s (t)

AUTOMATIC CIRCUITS
FOR WEIGHT
ADJUSTMENT

DESIRED
SIGNAL

d (t) +

Σ

e (t)
ERROR SIGNAL

−
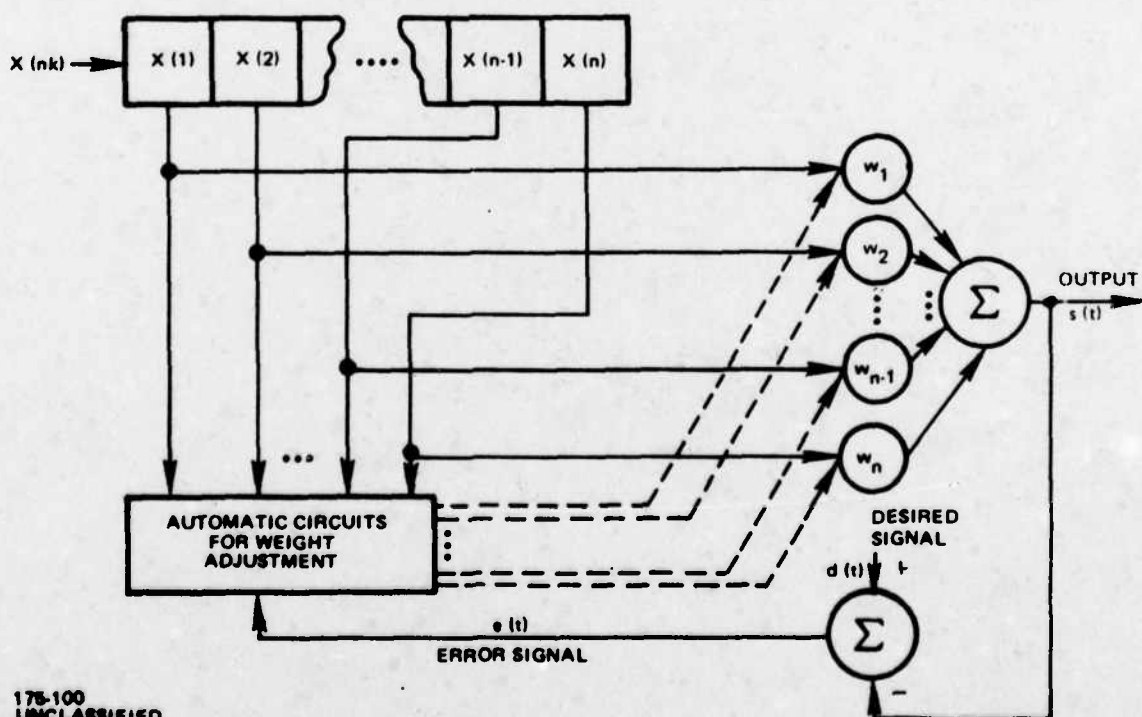
175-100
UNCLASSIFIED

Figure C-2. Block Diagram of Adaptive Equalizer

C-2

In Figure C-2, the heavy lines show the paths of signal flow, the dashed lines show functions related to weight changing or adaptation processes.

The output signal s(t) in Figure C-2 is the weighted sum

$$s(t) = \sum_{i=1}^{n} w_i x_i(t) \tag{1}$$

where n is the number of weights; or, using vector notation

$$s(t) = W^T X(t) \tag{1a}$$

where W^T is the transpose of the weight vector

$$W \triangleq \begin{bmatrix} w_1 \\ \cdot \\ \cdot \\ \cdot \\ w_i \\ \cdot \\ \cdot \\ \cdot \\ w_n \end{bmatrix} \tag{2}$$

and the signal input vector is

$$X(t) \overset{\triangle}{=} \begin{bmatrix} x_1(t) \\ \cdot \\ \cdot \\ \cdot \\ x_i(t) \\ \cdot \\ \cdot \\ \cdot \\ x_n(t) \end{bmatrix} \qquad (3)$$

For digital systems, the input signals are in discrete time sampled data form and the output is written

$$s(j) = W^T X(j) \qquad (4)$$

where the index j indicates the jth sampling instant.

In order that adaptation take place, a "desired response" signal, d(t) when continuous or d(j) when sampled, must be supplied to the adaptive element.

The difference between the desired response and the output response forms the error signal $\epsilon(j)$

$$\epsilon(j) = d(j) - W^T X(j) \qquad (5)$$

This signal is used as a control signal for the weight adjustment.

The purpose of the adaptation or weight changing processes is to find a set of weights that will permit the output response of the adaptive element at each instant of time to be equal to or as close as possible to the desired response. For each input signal vector $X(j)$, the error $\epsilon(j)$ of (5) should be made as small as possible.

Consider the finite set of linear simultaneous equations

$$
\begin{cases}
W^T X(1) = d(1) \\
W^T X(2) = d(2) \\
\quad \cdot \qquad \quad \cdot \\
\quad \cdot \qquad \quad \cdot \\
\quad \cdot \qquad \quad \cdot \\
W^T X(j) = d(j) \\
\quad \cdot \qquad \quad \cdot \\
\quad \cdot \qquad \quad \cdot \\
\quad \cdot \qquad \quad \cdot \\
W^T X(N) = d(N)
\end{cases}
\tag{6}
$$

where N is the total number of input signal vectors; each vector is a measurement of an underlying n-dimensional random process. There are N equations, corresponding to N instants of time at which the output response values are of concern; there are n "unknowns", the n weight values which form the components of W. The set of equations (6) will usually be overspecified and inconsistent, since in the present application, with an ample supply of input data, it is usual that $N \gg n$.

When N is very large compared to n, one is generally interested in obtaining a solution of a set of N equations [each equation in the form of (5)] which minimizes the sum of the squares of the errors. That is, a set of weights W is found to minimize

$$
\sum_{j=1}^{N} \epsilon^2(j)
\tag{7}
$$

When the input signals can be regarded as stationary stochastic variables, one is usually interested in finding a set of weights to minimize mean-square error. The quantity of interest is then the expected value of the square of the error; i.e., the mean-square error, given by

$$
E[\epsilon^2(j)] \triangleq \overline{\epsilon^2}
\tag{8}
$$

The set of weights that minimizes mean-square error can be calculated by squaring both sides of (5) which yields

$$
\epsilon^2(j) = d^2(j) + W^T X(j) X(j)^T W - 2d(j) W^T X(j)
\tag{9}
$$

and then taking the expected value of both sides of (9)

$$E[\epsilon^2(j)] = E[d^2 + W^T X(j) X^T(j)W - 2W^T d(j) X(j)]$$

$$= E[d^2] + W^T \Phi(x, x)W - 2W^T \Phi(x, d) \tag{10}$$

where

$$\Phi(x, x) \triangleq E[X(j) X^T(j)] \triangleq E \begin{bmatrix} x_1 x_1 & x_1 x_2 & \cdots & x_1 x_n \\ x_2 x_1 & & \cdots & x_2 x_n \\ \cdot & & & \\ \cdot & & & \\ \cdot & & & \\ x_n x_1 & & & x_n x_n \end{bmatrix} \tag{11}$$

and

$$\Phi(x, d) \triangleq E[X(j) d(j)] \triangleq E \begin{bmatrix} x_1 d \\ x_2 d \\ \cdot \\ \cdot \\ \cdot \\ x_i d \\ \cdot \\ \cdot \\ \cdot \\ x_n d \end{bmatrix} \tag{12}$$

The symmetric matrix $\Phi(x, x)$ is a matrix of cross correlations and autocorrelations of the input signals to the adaptive element, and the column matrix $\Phi(x, d)$ is the set of cross correlations between the n input signals and the desired response signal.

The mean-square error defined in (10) is a quadratic function of the weight values. The components of the gradient of the mean-square error function are the partial derivatives of the mean-square error with respect to the weight values. Differentiating (10) with respect to W yields the gradient $\nabla E[\epsilon^2]$, a linear function of the weights,

$$\nabla E[\epsilon^2] = 2\Phi(x, x) W - 2\Phi(x, d) \tag{13}$$

C-6

When the choice of the weights is optimized, the gradient is zero. Then

$$\Phi(x, x)W = \Phi(x, d)$$

$$W = \Phi^{-1}(x, x) \Phi(x, d) \qquad (14)$$

The optimum weight vector W is the one that gives the least mean-square error.

One way of finding the optimum set of weight values is to solve (14), which requires a matrix inversion. This solution presents serious computational problems when the number of weights n is large and when data rates are high. Fortunately, the matrix is Toeplitz, and an algorithm exists which can solve the simultaneous equations without a large amount of storage and with much less computation than required to solve (14) directly. This technique is called the Trench algorithm. A second approach is to minimize the mean-square error by one of a number of gradient-search techniques based on the method of steepest descent. One that has proven to be very useful is the LMS algorithm. Changes in the weight vector are made along the direction of the estimated gradient vector. Accordingly,

$$W(j + 1) = W(j) + k_s \nabla(j) \qquad (15)$$

where

$$W(j) \triangleq \text{weight vector before adaptation}$$

$$W(j + 1) \triangleq \text{weight vector after adaptation}$$

$$k_s \triangleq \text{scalar constant controlling rate of convergence and stability } (k_s < 0)$$

$$\nabla(j) \triangleq \text{estimated gradient vector of } \epsilon^2 \text{ with respect to } W.$$

The Trench algorithm and the LMS algorithm used for channel equalization are compared in Appendix D.

C.2   THE LMS ALGORITHM

The LMS algorithm obtains the estimated gradient of the mean-square error function by taking the gradient of a single time sample of the squared error, which for a real signal gives

$$\nabla(j) = \nabla[\epsilon^2(j)] = 2\epsilon(j)\nabla[\epsilon(j)]$$

From (5)

$$\nabla[\epsilon(j)] = \nabla[d(j) - W^T(j)X(j)] = -X(j)$$

Thus

$$\nabla(j) = -2\epsilon(j) X(j) \tag{16}$$

The gradient estimate of (16) is unbiased, as will be shown by the following argument. For a given weight vector $W(j)$, the expected value of the gradient estimate is

$$E[\nabla(j)] = -2E[\{d(j) - W^T(j) X(j)\} X(j)]$$

$$= -2[\Phi(x, d) - W^T(j) \Phi(x, x)] \tag{17}$$

Comparing (13) and (17), we see that

$$E[\nabla(j)] = \nabla E[\epsilon^2]$$

and therefore, for a given weight vector, the expected value of the estimate equals the true value.

Using the gradient estimation formula given in (16), the weight iteration rule (20) becomes

$$W(j + 1) = W(j) - 2k_s\epsilon(j) X(j) \tag{18}$$

and the next weight vector is obtained by adding to the present weight vector the input vector scaled by the value of the error.

The LMS algorithm is given by (18). It is directly usable as a weight-adaptation formula for digital systems. An equivalent differential equation which can be used in analog implementation of continuous systems is given by

$$W(t) = -2k_s \int_0^t \epsilon(\xi) X(\xi)d\xi \tag{19}$$

For the present communications system and equalization in the IF, both the received time samples $\underline{X}(j)$ and the weights $\underline{W}(j)$ are complex quantities. Therefore, the real and imaginary parts of the weights $\underline{W}(j)$ are driven separately to minimize the mean square error

$$\overline{e^2} = E\left[e(j)\, e^*(j)\right] \tag{20}$$

where $e^*(j)$ represents the complex conjugate of $e(j)$. Let subscript R denote the real part of a complex quantity (scalar or vector) and subscript I denote the imaginary part of that quantity. The squared error is given by

$$e^*(j)\, e(j) = s^*(j)\, s(j) + d^*(j)\, d(j) - s(j)\, d^*(j) - s^*(j)\, d(j) \tag{21}$$

Let $\nabla_R(j)$ denote the gradient of $e^*(j)\, e(j)$ with respect to the real part of $\underline{W}(j)$ and $\nabla_I(j)$ denote the gradient of $e^*(j)\, e(j)$ with respect to the imaginary part of $\underline{W}(j)$. These derivatives are

$$\underline{\nabla_R}(j) = -2\left(e_R(j)\, \underline{X_R}(j) + e_I(j)\, \underline{X_I}(j)\right) = -2\,\mathrm{Re}\left\{\underline{e}(j)\, \underline{X}(j)^*\right\}$$

$$\underline{\nabla_I}(j) = -2\left(e_I(j)\, \underline{X_R}(j) - e_R(j)\, \underline{X_I}(j)\right) = -2\,\mathrm{Im}\left\{\underline{e}(j)\, \underline{X}(j)^*\right\} \tag{22}$$

The subsequent weights are given by

$$\underline{W}(j+1) = \underline{W}(j) + k_s\left(\underline{\nabla_R}(j) + j\underline{\nabla_I}(j)\right) = \underline{W}(j) - 2\,k_s\,\underline{e}(j)\,\underline{X}(j)^* \tag{23}$$

An important consideration is the selection of a value for the convergence constant k_s. Convergence of the weight vector is assured if and only if the convergence constant is not too large. On the other hand, if k_s is too small, the weights will converge but at an exceedingly slow rate. Reference 24 derives the convergence criterion for k_s and shows that satisfactory convergence is obtained with

$$\frac{-1}{\sum\limits_{i=1}^{n} E[x_i^2]} \quad < k_s < 0 \tag{24}$$

When slow, precise adaptation is desired, k_s is usually chosen such that

$$\frac{-1}{\sum\limits_{i=1}^{n} E[x_i^2]} \quad \ll k_s < 0 \tag{25}$$

This convergence process of this algorithm produces an exponentially decreasing mean-square error implying a rapid initial decrease in error, slowing as adaptation continues. Generally, a good rate of stable convergence has been obtained with

$$\tag{26}$$

$$k_s = \frac{0.1}{\sum\limits_{i=1}^{n} E[x_i^2]}$$

C.3 THE TRENCH ALGORITHM

For complex samples, the direct solution of (14) for the optimum set of weights requires the inversion of a Hermitian Toeplitz matrix. (A matrix is Toeplitz if all elements along a given diagonal have the same value. A matrix is Hermitian if any two elements in symmetrical locations about the main diagonal have complex conjugate values.) For a complex representation, (14) may be written as

$$\sum_{i=1}^{n} w_i \, \rho_{m-i} = v_m ; \quad 1 \leq m \leq n \tag{27}$$

where

$$\rho_m = \sum_{k=1}^{N} x_k \, x_{k-m}^*$$

$$v_m = \sum_{k=1}^{N} d_k \, x_{k+m}^* \tag{28}$$

The Trench algorithm for solving (27) is now summarized. Let ρ_o be normalized to unity, and define the row vectors of order i

$$R_i = [\rho_1, \rho_2, .. \rho_i]$$

$$G_i = [g_1, g_2, .. g_i]$$

$$W_i = [w_1, w_2, .. w_i] \tag{29}$$

Define $\hat{R}_i$, $\hat{G}_i$ as row vectors with the elements in reversed order and the superscript T denotes the transponse. Initialize with

$$w_1 = v_1$$

$$g_1 = -\rho_1$$

$$\lambda_1 = 1 - \rho_1 \rho_1^* \tag{30}$$

The recursion is

$$\mu_i = (v_{i+1} - W_i \hat{R}_i^T)/\lambda_i \quad ; \quad 1 \leq i \leq M - 1$$

$$\beta_i = -(\rho_{i+1} + G_i \hat{R}_i^T)/\lambda_i \quad ; \quad 1 \leq i \leq M - 2$$

$$G_{i+1} = [G_i + \beta_i \hat{G}_i^*, \beta_i] \quad ; \quad 1 \leq i \leq M - 2$$

$$W_{i+1} = [W_i + \mu_i \hat{G}_i^*, \mu_i] \quad ; \quad 1 \leq i \leq M - 1$$

$$\lambda_{i+1} = \lambda_i(1 - \beta_i \beta_i^*) \quad ; \quad 1 \leq i \leq M - 2 \tag{31}$$

A listing of a FORTRAN subroutine of the Trench algorithm for solving a Hermitian Toeplitz set of equations is shown in Figure C-3. By removing the complex designations and the complex conjugates, the algorithm applies to a real symmetric Toeplitz set.

```
      SUBROUTINE CTREN(TOP,W,G,M)
C    TOP IS FIRST COLUMN OF HERMITIAN TOEPLITZ MATRIX,W IS VECTOR ON
C    RIGHT HAND SIDE AS AN INPUT AND IS THE SOLUTION VECTOR AS THE OUTPUT
C    ,G IS A WORK ARRAY,M IS THE NUMBER OF UNKNOWNS AS AN INPUT.
C    IN THE MAIN PROGRAM,TOP IS A COMPLEX ARRAY OF SIZE M,W IS A COMPLEX
C    ARRAY OF SIZE M,G IS A COMPLEX ARRAY OF SIZE 2M, M IS AN INTEGER.
C    THE TOEPLITZ MATRIX DOES NOT HAVE TO BE NORMALIZED TO BE UNITY ON
C    THE MAIN DIAGONAL.
      DIMENSION TOP(1),W(1),G(1)
      COMPLEX AMU,BETA,TOP,W,G
      K=M-1
      TOP1=TOP(1)
      DO 100 J=1,M
      TOP(J)=(TOP(J))/TOP1
  100 W(J)=W(J)/TOP1
      AL=1-TOP(2)*CONJG(TOP(2))
      G(1)=-TOP(2)
      DO 200 J=1,K
      AMU=W(J+1)
      DO 250 JJ=1,J
  250 AMU=AMU-TOP(J-JJ+2)*W(JJ)
      AMU=AMU/AL
      IF(J.EQ.K)GO TO 375
      BETA=-TOP(J+2)
      DO 300 JJ=1,J
  300 BETA=BETA-G(JJ)*TOP(J-JJ+2)
      BETA=BETA/AL
      DO 350 JJ=1,J
  350 G(JJ+M)=G(JJ)+BETA*CONJG(G(J-JJ+1))
  375 CONTINUE
      DO 400 JJ=1,J
  400 W(JJ)=W(JJ)+AMU*CONJG(G(J-JJ+1))
      W(J+1)=AMU
      IF(J.EQ.K)GO TO 200
      DO 450 JJ=1,J
  450 G(JJ)=G(JJ+M)
      G(J+1)=BETA
      AL=AL*(1-BETA*CONJG(BETA))
  200 CONTINUE
      RETURN
      END
```

Figure C-3.   FORTRAN Subroutine for Solving a Hermitian Toeplitz
Set of Simultaneous Equations

APPENDIX D
COMPARISON OF CONTROL ALGORITHMS
FOR ADAPTIVE CHANNEL EQUALIZATION

A channel used for data transmission suffers a degradation from intersymbol interference when the transfer function displays amplitude or phase distortion. A transversal filter with adjustable tap weights is a means for equalizing the overall channel response so as to reduce or eliminate the intersymbol interference. Considering only phase distortion, essentially perfect equalization can be achieved thereby. An algorithm to control the tap weights and force them to converge to an optimum solution is required. The general concept of such an algorithm is to compare the equalizer output with the desired signal to derive an error signal which contains the requisite control information.

The description above is couched in the framework of an iterative or adaptive algorithm; i.e., an iterative algorithm acts to drive the tap weights toward the set of optimum values which minimizes the error criterion (typically mean square error). Alternately, a noniterative algorithm can be derived which forms a set of simultaneous equations to be solved directly for the optimum weights. Comparison of the two approaches is of interest.

Since the error can be defined only when the error signal is known, the basic equalization procedure implies a training mode in which a known signal is transmitted over the channel. Alternatively, on-line equalization may be desired while data is being transmitted but is unknown prior to reception. A decision-directed mode is then utilized, wherein the desired signal is recreated from the output data. The feasibility of decision-directed equalization at low signal-to-noise ratios must be studied.

In the following, the channel is represented by samples taken at the bit rate. This implies a demodulator which tracks the actual bit timing and extracts samples synchronized to the bits. For purposes of computer simulations the composite effect of the channel filter and the demodulator integrate-and-dump detector is represented by the sampled impulse response. This will suffice for a comparative investigation of equalization algorithms, even though a more detailed channel simulation is still desirable.

The following is confined to baseband channels; however, all algorithms can be generalized to bandpass by merely introducing a complex representation.

D.1   NONITERATIVE EQUALIZATION WITH PERIODIC SIGNAL

Assume the desired signal x transmitted over the dispersive linear channel is periodic, so that the channel output y is also periodic (noise is not yet introduced). In terms of the impulse response h of the channel, the channel output is related to the input by the periodic convolution

$$y_m = \sum_{i=1}^{I} h_i \, x_{m-i} \tag{1}$$

where I is the duration of the impulse response. Now suppose the channel is equalized by a transversal filter with tap weights w, so that the equalizer output z is

$$z_k = \sum_{m=1}^{M} w_m \, y_{m+k} \tag{2}$$

also a periodic convolution. There are M weights in the filter. The error in the equalized output at time k with respect to a desired output d is

$$\epsilon_k = d_k - z_k \tag{3}$$

and the mean square error over the period K is

$$\epsilon^2 = \sum_{k=1}^{K} \left| d_k - z_k \right|^2 \tag{4}$$

which is a quadratic function of the tap weights. Hence, there is a single minimum of ϵ^2 as a function of the weights w.

Taking the above quantities as real (i.e., at baseband), we may differentiate ϵ^2 with respect to each tap weight w_i. Setting these derivatives to zero, a set of simultaneous equations in the unknown weights to minimize the error is obtained. Thus

$$\frac{\partial \epsilon^2}{\partial w_m} = -2 \sum_{k=1}^{K} \epsilon_k \frac{\partial z_k}{\partial w_m} = -2 \sum_{k=1}^{K} \epsilon_k y_{m+k} = 0 \tag{5}$$

The form of (5) indicates the optimum solution forces the average correlation between the instantaneous error and the channel output for various delays to be zero. One technique for solving the equations is to apply a steepest-descent type of algorithm. The LMS algorithm[24, 26] directly computes the instantaneous correlations between the error and the delayed channel outputs, and uses these quantities to drive the tap weights in a closed control loop (with an appropriate time constant) until the correlations approach zero as specified by (5). The modification of the LMS algorithm to complex quantities (i.e., a bandpass representation) is straightforward[27], and appears in Appendix C.

Substituting (2) and (3) into (5) and changing a subscript to eliminate duplication, the simultaneous equations in the unknown tap weights become

$$\sum_{i=1}^{M} \rho_{m-i} w_i = v_m \tag{6}$$

where

$$\rho_m = \sum_{k=1}^{K} y_k y_{m+k} \quad ; \quad \text{autocorrelation}$$

$$v_m = \sum_{k=1}^{K} d_k y_{m+k} \quad ; \quad \text{crosscorrelation} \tag{7}$$

Equation (6) is simplified by the fact that the matrix is symmetrical Toeplitz. If there are M unknown weights, the equation requires M values of the autocorrelation of the channel output y and M values of the crosscorrelation between the channel output y and the desired output d. An algorithm (the Trench algorithm) is available to solve the simultaneous equations without a large amount of storage and with much less computation than needed to solve a general set of simultaneous equations[28]. Figure D-1 is a listing of the FORTRAN subroutine TRENCH to solve the simultaneous equations by the Trench algorithm, and it can easily be generalized to handle the complex case. (See Appendix C.)

The solution of (6) is the set of optimum equalizer tap weights which minimizes the mean square error for the case of periodic signals. In the following, this equalization technique is called the Trench algorithm. An alternate computational approach attempts to minimize the maximum instantaneous error. A routine for Chebyshev approximation of an arbitrary function (defined at a set of points) as a linear sum of basis functions is available in the IBM Scientific Subroutine Package, and can be applied to this problem. However, for a large number of samples, the computation time is found to be excessive. On the other hand, the Trench algorithm can be utilized easily and economically even with as many as 60 unknown tap weights. (The upper limit for reasonable computation was not investigated further.)

The Trench algorithm is very simple to utilize in a computer simulation. The desired input signal x is generated (in this report it is a binary pseudorandom waveform) and stored in an array. The channel output y is computed by the periodic convolution (1) and stored in an array, and then the M autocorrelation and M cross-correlation values are computed by (7). The TRENCH subroutine then computes the set of optimum equalizer tap weights w. The desired output is defined to make the maximum tap weight occur roughly in the middle of the equalizer. The error between the desired output and the equalizer output can then be computed and plotted as a function of time. The desired output is a displacement of the input prior to distortion.

```
C     SUBROUTINE TRENCH(TOP,C,G,M)                                  TRE000210
C     TOP IS TOP ROW OF TOEPLITZ MATRIX,C IS VECTOR ON RIGHT HAND SIDE TRE000200
C     AS AN INPUT AND IS THE SOLUTION VECTOR AS THE OUTPUT,G IS A     TRE000300
C     WORK ARRAY,M IS THE NUMBER OF UNKNOWNS AS AN INPUT.            TRE000400
C     IN THE MAIN PROGRAM,TOP IS A REAL ARRAY OF SIZE M,C IS A REAL ARRAY TRE000500
C     OF SIZE M,G IS A REAL ARRAY OF SIZE 2M, M IS AN INTEGER.       TRE000600
C     THE TOEPLITZ MATRIX DOES NOT HAVE TO BE NORMALIZED TO BE UNITY ON TRE000700
C     THE MAIN DIAGONAL.                                            TRE000800
      DIMENSION TOP(1),C(1),G(1)                                    TRE000900
      K=M-1                                                         TRE001000
      TOP1=TOP(1)                                                   TRE001100
      DO 100 J=1,M                                                  TRE001200
100   TOP(J)=C(J)/TOP1                                              TRE001300
      AL=1.-TOP(2)**2                                               TRE001400
      G(1)=-TOP(2)                                                  TRE001500
      DO 200 J=1,K                                                  TRE001600
      TH=C(J+1)                                                     TRE001700
      DO 250 JJ=1,J                                                 TRE001800
250   TH=TH-TOP(J-JJ+2)*C(JJ)                                       TRE001900
      AMU=TH/AL                                                     TRE002000
      IF(J.EQ.K)GO TO 375                                           TRE002100
      GAMMA=-TOP(J+2)                                               TRE002200
      DO 300 JJ=1,J                                                 TRE002300
300   GAMMA=GAMMA-G(JJ)*TOP(J-JJ+2)                                 TRE002400
      BETA=GAMMA/AL                                                 TRE002500
      DO 350 JJ=1,J                                                 TRE002600
350   G(JJ+M)=G(JJ)+BETA*G(J-JJ+1)                                  TRE002700
375   CONTINUE                                                      TRE002800
      DO 400 JJ=1,J                                                 TRE002900
400   C(JJ)=C(JJ)+AMU*G(J-JJ+1)                                     TRE003000
      C(J+1)=AMU                                                    TRE003100
      IF(J.EQ.K)GO TO 200                                           TRE003200
      DO 450 JJ=1,J                                                 TRE003300
450   G(JJ)=G(JJ+M)                                                 TRE003400
      G(J+1)=BETA                                                   TRE003500
      AL=AL*(1-BETA**2)                                             TRE003600
200   CONTINUE                                                      TRE003700
      RETURN                                                        TRE003800
      END                                                           TRE003900
                                                                    TRE004000
```

Figure D-1. FORTRAN Subroutine for TRENCH Algorithm

Figure D-2 is an illustration of the results. The channel impulse response has the values .75, 0, 1, 0, -.75, which is distortion due to a single pair of anti-symmetric paired echoes. This corresponds to phase distortion in the channel. Note that the distortion exceeds unity (the "eye" pattern with binary transmission does not have an open region). Figure 2 plots the error after 32-tap equalization with a period of 8192 samples. The peak error relative to a desired binary input waveform of unit amplitude is seen to be .015; thus, the equalization by the Trench algorithm is very accurate for this channel despite the excessive phase distortion.

D. 2   <u>NONITERATIVE EQUALIZATION WITH NONPERIODIC SIGNAL</u>

To enable automatic channel equalization, a known signal can be trans-mitted. Such a signal is typically periodic, and the above theory is directly applicable. However, to track subsequent time variations in the channel, an adaptive mode is needed which can work with a nonperiodic signal. An iterative algorithm like the LMS algorithm is natural for such adaptation to track slow changes of the channel distortion.

To utilize the noniterative Trench algorithm for adaptation in the non-periodic case, the autocorrelation and crosscorrelation values of (7) can be calculated for a finite length "snapshot" of the channel output. Provided the snapshot is much longer than the equalizer span, the error due to the edge effect can be made small; however, the channel must remain essentially fixed over the snapshot. Thus, a trade-off exists between accuracy of equalization and speed of adaptation, in similarity to choice of time constant for the LMS algorithm.

For the postulated dispersive channel with an antisymmetric pair of echoes, the accuracy of equalization by the Trench algorithm was investigated empirically by selecting different snapshot durations. Figure D-3 is a plot of the results for the peak error after equalization. The signal was generated with a period of 8192 samples, and the snapshot duration was a fraction of this period. After computing the equalizer tap weights from the snapshot, the error for each of the 8192 samples was calculated. Two different channel impulse response cases were assumed:
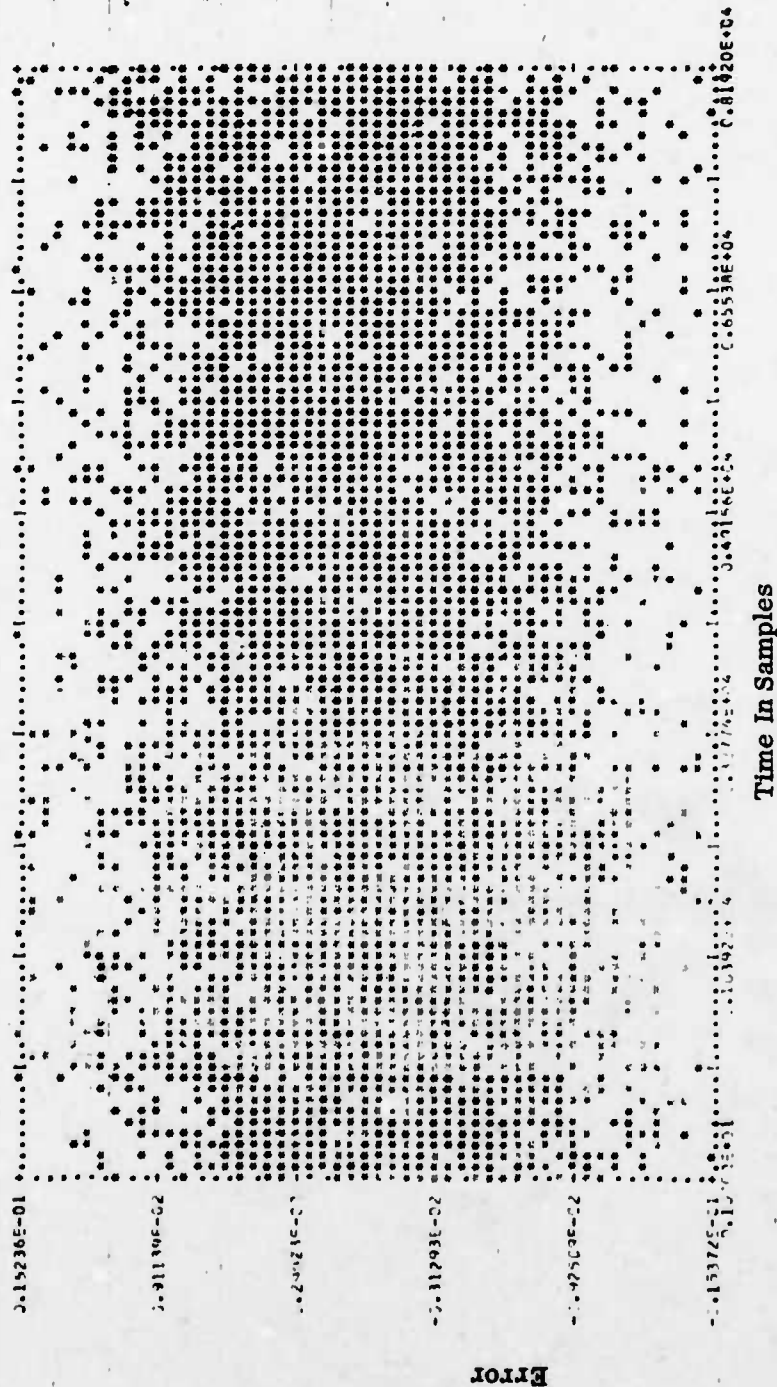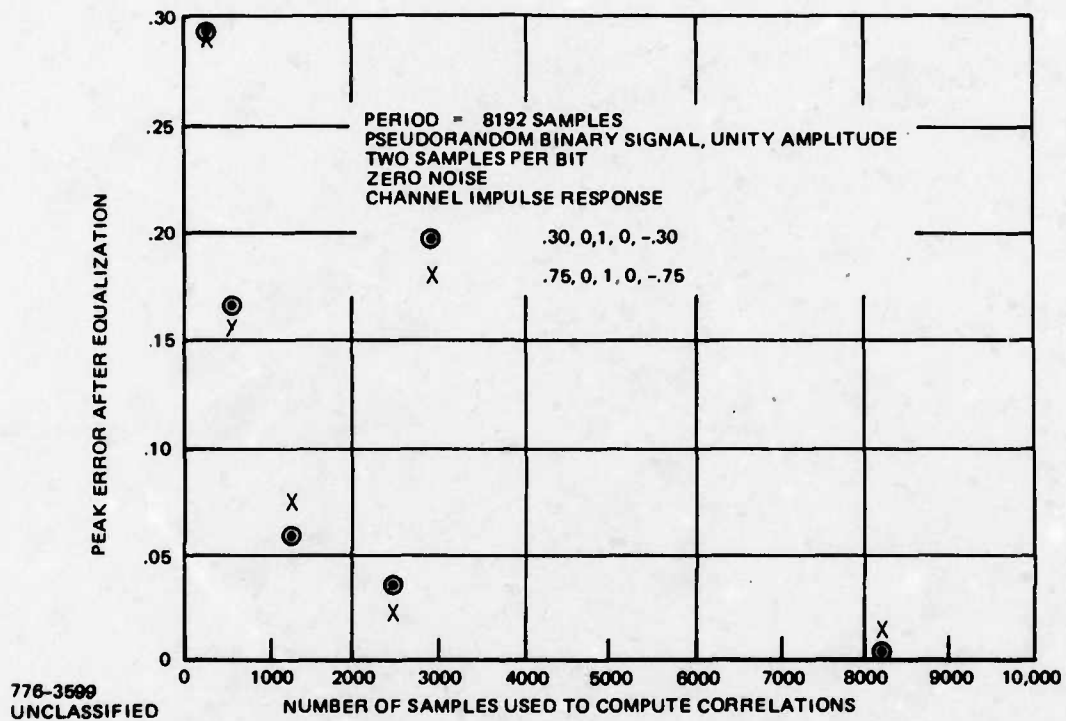
Figure D-2. Plot of Error After Equalization, 32 Taps

Time In Samples

Error

Figure D-3. Equalization by Trench Algorithm

$$.3, \ 0, \ 1, \ 0, \ -.3$$

$$.75, \ 0, \ 1, \ 0, \ -.75$$

the latter having a distortion exceeding unity.

Inspection of Figure D-3 indicates that excellent equalization of both cases is achieved for snapshot durations in excess of about 2000 samples. Other data, not plotted, shows a similar behavior when these are four samples per bit of the pseudo-random binary signal (i.e., a reduced signalling bandwidth).

D.3   <u>NONITERATIVE EQUALIZATION OF NOISY SIGNAL</u>

If the channel output is received at a low signal-to-noise ratio, equalization will be affected by the noise. Nevertheless, a minimum mean square equalization algorithm still accomplishes the desired objective, except that the algorithm acts to minimize the sum of the errors due to both noise and channel distortion.

To investigate the effect of additive noise, the computer simulation generates Gaussian noise samples which are independent from sample to sample. In other words, the channel output is

$$y_m = \sum_{k=1}^{K} h_k x_{m-k} + n_m \tag{8}$$

where n_m is an independent Gaussian sample. To equalize by the Trench algorithm, the correlations are still computed by (7) over the selected snapshot duration; however, now the noise will perturb these correlation values. To investigate the effect of this perturbation when the simultaneous equations are solved to yield the tap weights, the peak distortion after equalization is computed by passing the channel output through the equalizer without the additive noise. (This is easy to do in a computer simulation.) The noise power after equalization is obtained by summing the squares of the tap weights, since the noise samples are independent.

Table D-I presents some results for equalization of a noisy signal by the Trench algorithm. Note that increasing the snapshot duration beyond 1000 samples does not significantly reduce the resultant error, at the assumed low signal-to-noise ratio.

Snapshot Duration	Peak Distortion	RMS Noise
1024 samples	0.344	0.29
2048	0.334	0.29
8192	0.295	0.29

Period = 8192 samples
2 samples/bit
Channel impulse response
= .75, 0, 1, 0, -.75
rms noise/rms signal
= 0.34 before equalization

Table D-I. Equalization of Noisy Signal

To show the tradeoff between distortion and noise after equalization, we can compute the equalizer tap weights from the correlation values which would be measured for a snapshot of infinite duration. Since independent noise samples are added in the channel output, only the autocorrelation for $\tau = 0$ in (7) is affected by the noise in this case. Results are presented in Table D-II to show the peak distortion and the rms noise after equalization, as a function of the received signal-to-noise ratio. Note that the peak distortion and the rms noise after equalization are roughly equal at all signal-to-noise ratios. In other words, the mean square error tends to be minimized when the equalizer balances the resultant distortion and noise.

Before Equalization	After Equalization	
RMS Noise / RMS Signal	Peak Distortion	RMS Noise
0.34	0.26	0.29
0.17	0.13	0.16
0.068	0.051	0.070
0.034	0.026	0.036

Period = 8192 samples
2 samples/bit
Channel impulse response
= .75, 0, 1, 0, -.75

Table D-II. Equalization of Noisy Signal - Infinite Averaging Time

Minimization of the total mean square error is a practical concept in the sense that separation of noise and distortion is not possible in an actual equalizer, in contrast to a computer simulation. The question arises whether optimization of the total mean square error also optimizes other performance indices more meaningful for data transmission. As a test, the average probability of error for binary transmission at a rate of one bit per sample was computed when the channel was equalized at different noise levels by the Trench algorithm with infinite averaging. The computer simulation allowed the noise level for data transmission to differ from that assumed

for equalization. For all channel distortions and noise levels tested, it was found that the average probability of error actually was minimized by equalizing at the noise level assumed for data transmission. Apparently, the criterion of minimizing the mean square sum of distortion and noise also optimizes data demodulation (see Appendix E). The optimum is quite broad, however, in the sense that the average probability of error changes very slowly with the noise level assumed for equalization.

D.4   COMPARISON WITH EQUALIZATION VIA LMS ALGORITHM

It has been found in the above that to achieve accurate equalization by the Trench algorithm requires a fairly large number of samples, in excess of 2000. It is of interest to compare these results with results for equalization by the LMS algorithm with comparable adaptation times. The LMS algorithm controls the tap weights proportional to the correlation of instantaneous error with channel output, as an approximation to the average gradient of (5). Thus, for the LMS algorithm, the adaptation at the mth sample is

$$\Delta w_i = 2k \epsilon_m y_{m+i} \tag{9}$$

where k adjusts the rate of adaptation (i.e., time constant) to drive the error towards the minimum. There is an upperbound on k for stability.

The LMS algorithm was simulated initializing the tap weights all to zero. Figures D-4 and D-5 plot the absolute value of the error as a function of time for two different rates of adaptation. The channel impulse response is .75, 0, 1, 0, -.75, and there are two samples per bit of the pseudorandom binary input signal. With the faster rate of convergence, the error is reduced to less than .04 after roughly 3000 samples. While not quite as good as the Trench algorithm for this number of samples, see Figure D-3, the difference in results between the two algorithms is not of practical significance.
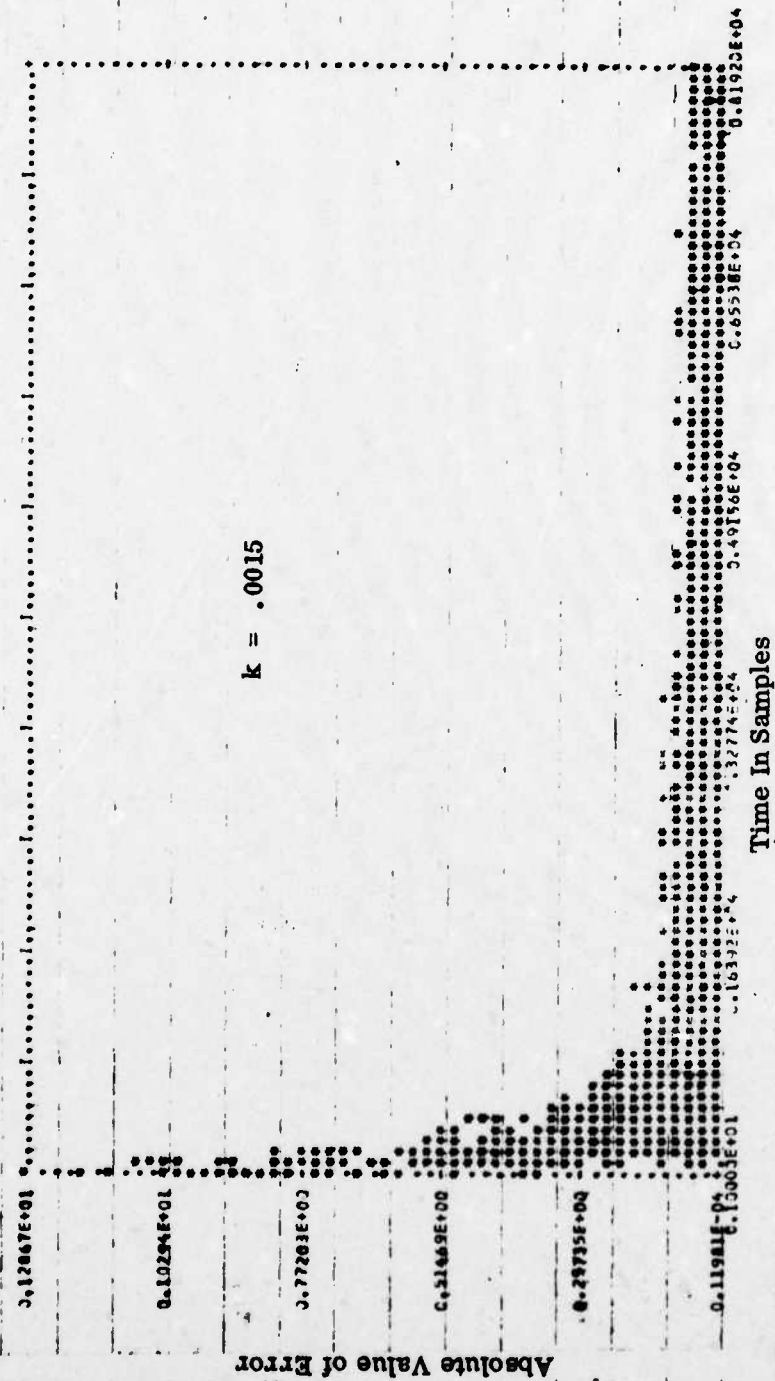
$k = .005$

Time In Samples

Absolute Value of Error

Figure D-4. LMS Algorithm Adaptation, 32 Taps

Figure D-5. LMS Algorithm Adaptation, 3² Taps

Now, let us consider a noisy received signal; where the channel output is specified by (8) with independent Gaussian noise samples. Figure D-6 plots the absolute value of distortion (i.e., noise not included) as a function of time, with adaptation on the noisy received signal at a relatively low signal-to-noise ratio. The rate of adaptation is a tenth of that employed for Figure D-4. Comparing Figure D-6 with Table D-I, which is for the same signal-to-noise ratio, it appears that there is no practical difference between the two algorithms with respect to the accuracy of the resultant equalization. Also, the rms noise after equalization is the same for both algorithms.

D. 5   DECISION-DIRECTED EQUALIZATION

The preceding discussion has assumed throughout that the desired signal is fully available to the equalization algorithm. Such is the case if a known signal is transmitted over the channel for purposes of "training" the equalizer. A training signal is typically generated as a pseudorandom sequence which is exactly reproducible in the receiver. With the aid of the training signal, either the LMS algorithm or the Trench algorithm achieves automatic equalization by solving for the tap weights of the transversal filter which minimizes the mean square error between the equalizer output and the known desired signal.

A common situation is that the channel introduces a relatively small amount of distortion, and data can be transmitted correctly except that the required E_b/N_o is excessive due to the intersymbol interference. The objective of equalization is to remove the distortion, which may be time varying, and enable operation at a reduced E_b/N_o. Furthermore, the equalization is to be accomplished on line while transmitting data without recourse to a special training signal. This mode of operation may be defined as adaptive equalization. The approach is to recreate the desired signal from the output bit decisions, on the assumption that the error rate is low when the equalization process is initiated. This is decision-directed adaptive equalization. (If necessary, a training signal could be transmitted for start up in a channel displaying severe distortion. After the preliminary automatic equalization, subsequent changes would be tracked in the decision-directed adaptive mode.)

If an error-correcting code is present, a choice exists between decision-directed operation either with raw bit decisions or with output bits after decoding. The latter are much more reliable, but the equalizer then must incorporate storage to compensate for the additional decoding delay (typically about 32 data bits with a Viterbi decoder, but possibly much greater if interleaving is utilized). Note also the need for a duplicate encoder to construct a replica of the signal actually transmitted over the channel.

RMS Noise / RMS Signal = 0.34

k = .0005

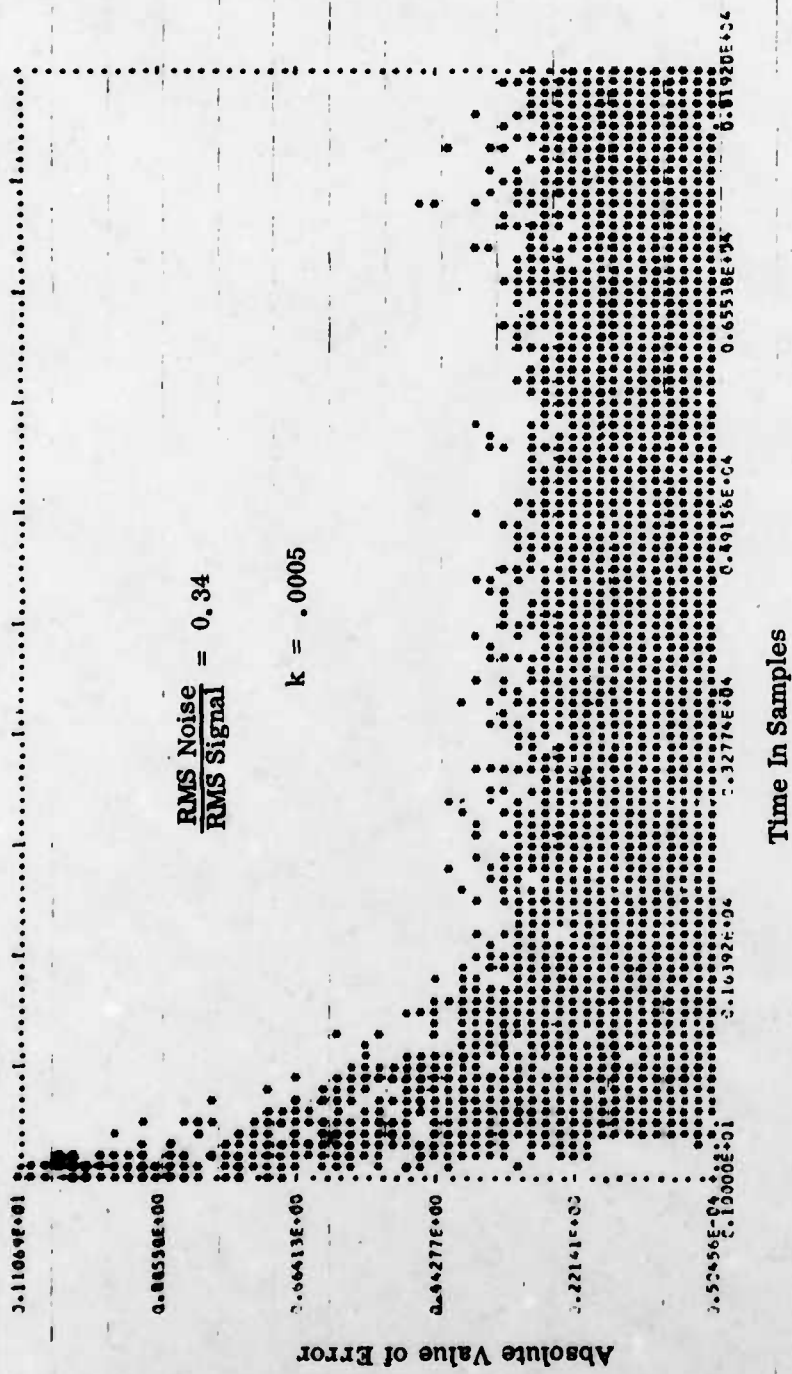Time In Samples

Absolute Value of Error

Figure D-6. LMS Algorithm Adaptation on Noisy Signal, 32 Taps

The modification of the theory above is to define the decision-directed error by

$$\hat{\epsilon}_k = \hat{x}_{k-d} - z_k \tag{10}$$

where $\hat{x}$ denotes the decision on the desired signal and d is an arbitrary shift. Then, $\hat{\epsilon}$ replaces ϵ in (5) and (9). Equation (7) becomes

$$\rho_m = \sum_{k=1}^{K} y_k \, y_{m+k}$$

$$\hat{v}_m = \sum^{K} \hat{x}_{k-d} \, y_{m+k} \tag{11}$$

to compute the correlation values for the decision-directed Trench algorithm. The decision-directed LMS algorithm is described by the adaptation

$$\Delta w_i = 2k \hat{\epsilon}_m \, y_{m+i} \tag{12}$$

The effect of decision errors must be investigated further.

Decision-directed operation destroys the unimodal character (i.e., unique optimum) of the quadratic error surface associated with a training mode. As the simplest example, observe that two equally good solutions exist with decision-directed operation, one of which simply inverts the data in similarity with a 180° phase reversal in tracking carrier phase. Thus, we must be concerned with possible existence of other solutions which are locally optimum but do not achieve the minimum error. These are a problem only if the equalizer is likely to be trapped in a local rather than a global optimum.

We now analytically investigate solutions to decision-directed equalization for a simple example. Let the impulse response of the channel be

$$h_1 = a$$
$$h_2 = 1$$
$$h_3 = -a \tag{13}$$

and assume a three-tap equalizer with weights w_1, w_2, w_3. By symmetry, $w_3 = -w_1$ in any solution; hence, there are two unknowns, w_1 and w_2. We assume the transmitted signal is a random binary sequence (± 1) and take the ensemble average in evaluating (11). Two equations in the two unknowns are needed. Now, from (1)

$$y_4 = ax_3 + x_2 - ax_1$$

$$y_5 = ax_4 + x_3 - ax_2$$

$$y_6 = ax_5 + x_4 - ax_3 \tag{14}$$

and from (11) with correct decisions

$$a_0 = E\left\{y_5^2\right\} \quad = 1 + 2a^2$$

$$a_1 = E\left\{y_5\, y_6\right\} = 0$$

$$v_0 = E\left\{x_3\, y_6\right\} = -a$$

$$v_1 = E\left\{x_3\, y_5\right\} = 1 \tag{15}$$

From (6), the simultaneous equations are

$$a_0 w_1 + a_1 w_2 = v_0$$

$$a_1 w_1 + a_0 w_2 = v_1 \tag{16}$$

and substituting from (15), the solution is

$$w_1 = -w_3 = -a/(1 + 2a^2)$$

$$w_2 = 1/(1 + 2a^2) \tag{17}$$

From (2), the equalizer output is

$$z_7 = w_1 y_6 + w_2 y_5 + w_3 y_4$$

$$= x_3 - \frac{a^2}{1 + 2a^2}\, (x_1 + x_5) \tag{18}$$

Note that for any value of a, the polarity of the equalized output z_7 gives the correct decision on x_3 regardless of the polarities of the nearby bits x_1 and x_5. The interpretation of this result is that decision-directed equalization by either the Trench algorithm or the LMS algorithm will be successful provided that correct decisions can be made prior to equalization.

When the channel distortion is large, the problem arises that correct decisions are not always made. Prior to equalization, the polarity of y_m is taken as the decision on x_{m-d}; with the LMS algorithm, this means initializing the tap weights with the center tap unity and all others zero. Then, if $a > 0.5$, (14) shows that an incorrect decision on x_3 will be made from y_5 as a consequence of the intersymbol interference when $x_4 = -1$ and $x_2 = +1$. We have to evaluate (11) for this case. The quantities a_0 and a_1 are unchanged from (15), b

$$\hat{v}_0 = E\left\{\hat{x}_3 \, y_6\right\}$$

$$\hat{v}_1 = E\left\{\hat{x}_3 \, y_5\right\} \tag{19}$$

where $\hat{x}_3$ is the decision on x_3, according to

$$\hat{x}_3 = \text{sign}(y_5) \tag{20}$$

Table D-III categorizes all bit sequences with $x_3 = +1$ and the decision errors if $a > 0.5$. By symmetry, all sequences with $x_3 = -1$ will yield the same average. We see from Table D-III

$$\hat{v}_0 = (1 - a)/2$$

$$\hat{v}_1 = (1 + 2a)/2 \tag{21}$$

Solving (16) with $\hat{v}_0$ and $\hat{v}_1$ substituted gives

$$w_1 = -w_3 = (1 - a)/(2 + 4a^2)$$

$$w_2 = (1 + 2a)/(2 + 4a^2) \tag{22}$$

x_2 x_3 x_4 x_5	y_5	y_6	$\hat{x}_3$	$\hat{x}_3\, y_5$	$\hat{x}_3\, y_6$	
+ + + +	1	1	+	1	1	
+ + + −	1	1 − 2a	+	1	1 − 2a	
+ + − +	1 − 2a	−1	−	−1 + 2a	1	← error
+ + − −	1 − 2a	−1 − 2a	−	−1 + 2a	1 + 2a	← error
− + + +	1 + 2a	1	+	1 + 2a	1	
− + + −	1 + 2a	1 − 2a	+	1 + 2a	1 − 2a	
− + − +	1	−1	+	1	−1	
− + − −	1	−1 − 2a	+	1	−1 − 2a	
Sum				4 + 8a	4 − 4a	

Table D–III. Tabulation of Intersymbol Interference Effect on Decision-Directed Equalization, when $a > 0.5$.

From (2), the equalized output is

$$z_7 = w_1\, y_6 + w_2\, y_5 + w_3\, y_4$$

$$= 0.5x_3 + 0.5(-x_2 + x_4) + \frac{a - a^2}{2 + 4a^2}\,(x_1 + x_5) \tag{23}$$

It is seen that the equalized output z_7 will yield an incorrect decision on x_3 if $x_4 = -1$ and $x_2 = +1$, as was presumed in obtaining the solution (22).

The existence of two different solutions to the decision-directed equalization problem has now been demonstrated for a simple example of channel distortion. The unwanted solution will be obtained only if the channel distortion is sufficiently high that incorrect decisions are made due to intersymbol interference. The LMS algorithm offers a conceptual way out of the difficulty by being compatible with use of a training mode to provide an initial degree of equalization such that the residual distortion is not large enough to cause incorrect decisions in the subsequent decision-directed adaptation process. It is expected, also, that using decisions after error correction decoding will enable tolerating large channel distortions.

D.6   DECISION-DIRECTED EQUALIZATION WITH NOISE

Decision-directed equalization when the signal-to-noise ratio is low raises the question as to the effect of decision errors due to noise. In particular, note that the probability of error is largest for those bits most subject to cancellation by inter-symbol interference. Some empirical results of computer simulations are now presented, where decision-directed equalization by the LMS algorithm, described by (12), is applied to reduce the distortion. The simulation assumes one sample per bit of random binary data and independent additive Gaussian noise samples. The probability of error after equalization is computed from the decision amplitude for each bit and averaged over the full period.

When the distortion exceeds unity prior to equalization so that decision errors are caused by intersymbol interference, as seen in Table D-III, it is found that the LMS algorithm can actually be trapped in the incorrect solution at high signal-to-noise ratios, when the equalizer is initialized with the center tap weight at unity and all others at zero. However, at low signal-to-noise ratios, the incorrect solution does not persist, and the equalizer converges to the correct solution. Thus, empirically, an alternative to a special training mode is available to handle the case of large channel distortion. Initially, additional noise is added sufficient to eliminate the trap in the incorrect solution of the decision-directed LMS algorithm. After a short time, the equalizer will have reduced the distortion sufficiently that the additional noise is no longer necessary to eliminate the trap and can be removed, after which convergence to the correct solution takes place.

To see how well the decision-directed equalizer works at a low signal-to-noise ratio, a comparison of probability of error results may be made to the results of equalization in a training mode. In this simulation, there is one sample per bit with additive independent Gaussian noise, and the channel distortion is described by the impulse response .75, 1, -.75. The decision-directed LMS algorithm is iterated for a period of 4096 bits with $k = .0005$ in (12), after which the average probability of error is computed over the period for the resulting tap weights at the given signal-to-noise ratio. For comparison, the equalizer tap weights is obtained for the same signal-to-noise ratio by the Trench algorithm for infinite averaging time in a training mode with correct decisions, and the average probability of error is again computed over the period. Results are shown in Figure D-7. The degradation due to the presumed anti-symmetric distortion, which corresponds to phase distortion in the channel, is seen to
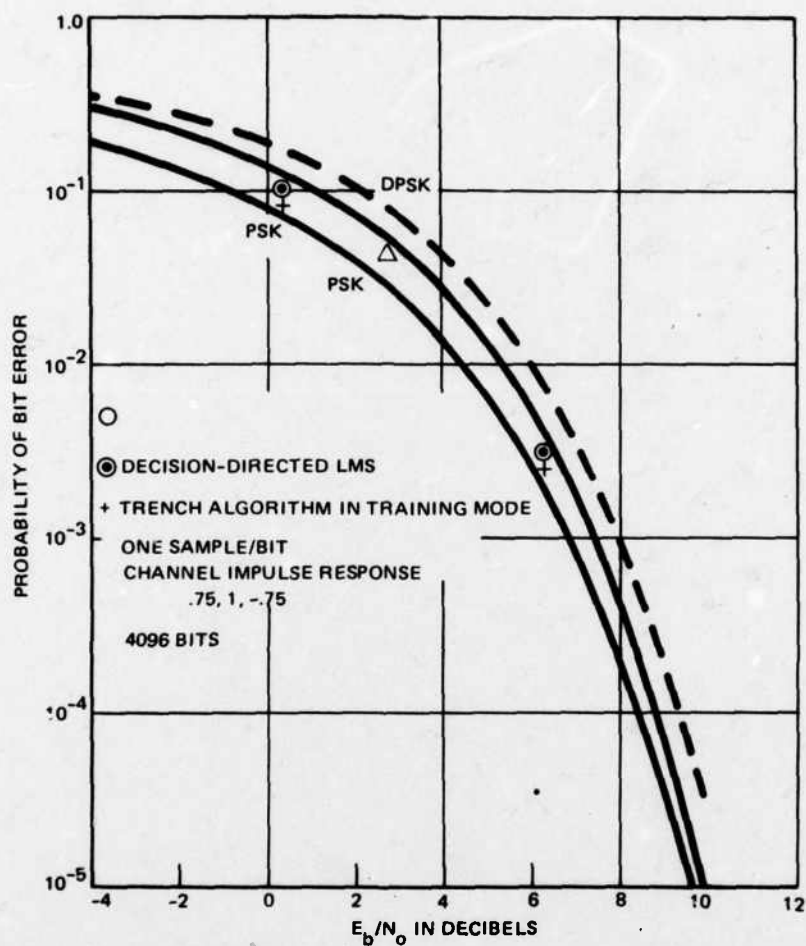
Figure D-7. Comparison of Decision-Directed Equalization
          With Training Mode

be very small after equalization. Furthermore, the decision-directed mode works well even at a relatively high average error rate (10 percent) for the bit decisions.

D.7   CONCLUSIONS

Applied to the problem of baseband equalization of a channel displaying phase distortion, a transversal filter is found to be very effective. A key question is how to control or compute the tap weights to optimize data transmission performance.

Two algorithms were studied. The first, which we call the Trench algorithm, calculates the autocorrelation function of the channel output and the cross-correlation of the channel output with the desired signal. The tap weights are then found by solving a set of simultaneous equations. Since the matrix is Toeplitz, the solution is quite economical of computer CPU time and storage, even when the number of taps is large. The second is the LMS algorithm, a type of steepest-descent algorithm, which iteratively converges on the solution to minimize the mean square error. Empirically, the two algorithms appear to have comparable performance, measured by resulting accuracy of equalization, when both process the same span of channel output signal (snapshot duration).

In a training mode with a periodic signal, the Trench algorithm is straight-forward, very accurate, and economical of computer resources. It is an "off-line" algorithm, however. The LMS algorithm appears more suited to a decision-directed adaptive mode, since even a rather inaccurate initial equalization will suffice to obtain good decisions for the subsequent adaptation process. Furthermore, the LMS algorithm has the conceptual virtue of applying feedback control to reduce the error in the structure actually being employed "on-line" for equalization.

Empirical studies of data channel error rate performance with binary data show excellent equalization by the decision-directed LMS algorithm even at the relatively high error rate of 10 percent.

No consideration was given to the problem of bit synchronization as it pertains to the comparison of desired signal with equalizer output.

APPENDIX E
A TECHNIQUE TO COMPUTE THE EQUALIZER TAP WEIGHTS
WHICH MINIMIZE QPSK PROBABILITY OF ERROR

This appendix describes a computational technique to obtain
the set of tap weights for the transversal filter equalizer which minimizes the
probability of error for QPSK at a specified E_b/N_o. It is intended to compare
performance using this optimum equalizer with results for an adaptive
equalizer controlled to minimize the decision-directed mean square error.
The latter control technique is feasible in an actual receiver with on-line data
transmission; hence, the previous results are indicative of performance achievable
in an actual data transmission system. Hopefully, there will be little difference.

The approach is to develop a computer simulation program which
returns the probability of error as a function of the set of tap weights. Then,
an available computer subroutine can be invoked which minimizes a function of
several real variables without requiring evaluation of function derivatives.

The approach to be developed allows the equalizer to be in the
receiver, in the transmitter, or even in both. Also, the transmitter and/or the
satellite channel could be allowed to saturate. However, the present discussion
studies the optimization of a receive equalizer for a linear channel with QPSK
modulation.

E.1   CRITIQUE OF MEAN SQUARE ERROR MINIMIZATION

Although the performance results based on controlling the equalizer tap
weights to minimize the mean square error appear tc have been very satisfactory,
this does not necessarily mean the optimum performance has been found. For
example, the error on a bit decision sample is treated as equally bad whether it
is a positive or a negative deviation from the desired hard decision value of ± 1.0.
In contrast, probability of error for PSK is minimized when each bit decision
sample is as large as possible*, with a fixed output noise power.

---

*This statement presumes antipodal binary modulation (i.e., BPSK, QPSK, or SQPSK)
with the decision boundary at zero. The duobinary technique discussed in Section 9
would require a modification to this statement to reflect the ternary decision.

Thus, compared to the solution minimizing the mean square error, improved performance should, in principle, be attained by readjusting the tap weights to increase the mean square error somewhat, provided that the distances to the decision boundary are thereby maximized. Since the equalizer also affects the output noise power, the distance to the decision boundary has to be normalized by the square root of the output noise power.

E.2   PERFORMANCE MEASURE

The probability of error is computed for a periodic data sequence with 32 data bits in the period and 32 samples per bit in the representation. For QPSK, there are two parallel data streams forming, respectively, the real and the imaginary parts of the signal; hence, 64 bits total. Phase distortionless filters having a 5-pole Butterworth amplitude characteristic are assumed. The transmitted signal is filtered and renormalized, so that E_b/N_o is defined with respect to the signal power actually received. The received signal is filtered again in the receiver, and the filter output is passed through an adjustable transversal filter equalizer defined by the set of complex tap weight values and the tap spacing.

The equalizer output is sampled at the bit rate, the clock phase being a parameter. For zero-phase filters, the sampling time is set at the midpoint of the bit*. Thus, 64 bit decision amplitudes are obtained by this sampling process (32 real parts and 32 imaginary parts). These amplitudes are denoted by A_k, and are a function of the equalizer tap weights. For an ideal QPSK system, these amplitudes would be ± 1.

Next, the output noise power is computed by a numerical integration in the frequency domain. The transfer function is that of the receive filter in cascade with the equalizer. Thus, for white noise, the normalized output noise power is

$$\sigma^2 = T_b \int_{-\infty}^{\infty} |H(j\omega)|^2 \left| \sum_m w_m e^{-j\omega mT} \right|^2 df \qquad (1)$$

where w_m denotes the mth complex tap weight of the equalizer, T_b is the bit duration, T is the tap spacing, and $H(j\omega)$ is the transfer function of the receive filter**. For an ideal QPSK system, $\sigma = 1$. Note that σ in (1) is a function of the tap weights.

---

*For a representation with an even number of samples per bit, the exact midpoint is not attainable.

**In (1), ω is the deviation from the center frequency.

The avarage probability of error P_e at a specified E_b/N_o is computed over the periodic data sequence for a Gaussian distribution of noise; hence,

$$P_e = (1/64) \sum_{k=1}^{64} .5 \, \text{erfc} \, [\sqrt{E_b/N_o} \, A_k \, D_k/\sigma] \qquad (2)$$

where D_k is the correct polarity for the kth bit decision. Note that P_e is a function of the tap weights through the A_k and σ, and implicitly is a function of the sampling clock phase.

In the sense of nonlinear programming, P_e is the objective function to be minimized. The variables are the real and imaginary parts of the tap weights, and the minimization is unconstrained*. Note that noise is not actually added to the signal in this computational approach.

E.3   MINIMIZATION PROCEDURE

The evaluation of (2) is done in a computer simulation program which is called as a subprogram by an available FORTRAN subroutine for function minimization. Subroutine CDMIN, written at the University of Waterloo, searches to find the minimum, starting at a specified set of initial values for the variables. This subroutine is particularly useful because it does not require evaluation of function derivatives. It uses the method of conjugate directions; i.e., by searching in various directions (in the multidimensional space of the variables) until the maximum points along those directions are encountered, improved directions to search can be developed.

The search is terminated by the subroutine either after a fixed number of iterations or after a specified accuracy for the values of the variables has been achieved. The value of the function minimum and the values of the variables at the minimum point are read out by the subroutine.

E.4   COMPARATIVE EXAMPLE

To illustrate the procedure, equalization of a narrowband channel was performed. The filters are zero phase with a 5-pole Butterworth amplitude characteristic. The transmit bandpass and receive bandpass are each $1/T_b$, where T_b is the bit duration on in-phase or quadrature for the QPSK signal.

---

*Since (2) is invariant to an overall gain factor, one variable could actually be held fixed.

The equalizer has 7 complex taps; thus, there are 14 variables affecting the value of (2). Unequalized, the probability of error for QPSK is 4.8×10^{-5} at $E_b/N_o = 10$ dB; this is a degradation of 1.2 dB from ideal QPSK.

Subroutine CDMIN was invoked and set to terminate after four iterations or after reaching an accuracy of 0.01 for the variables. The search terminated after three iterations, and the resulting probability of error is 7.7×10^{-6}, a degradation of 0.3 dB from ideal QPSK. Evaluation of (2) was done 110 times by the subroutine to find this minimum point. The sampling clock phase was held at the midpoint of the bit for all evaluations.

For comparison, the same QPSK channel was equalized by the LMS algorithm working to minimize the mean square error in the bit decision amplitudes at $E_b/N_o = 10$ dB. In this technique, noise is added to the received signal during the adaptation, but after convergence, the probability of error is computed using (2). The sampling clock place was held at the midpoint of the bit.

Convergence by the LMS algorithm was observed after 18 periods of the signal (contrast with 110 for CDMIN). The resulting probability of error is 8.2×10^{-6}, which is only slightly higher than achieved by CDMIN.

E.5   CONCLUSIONS

For the example tested of a narrowband channel, equalization by the LMS algorithm to minimize the mean square error yielded a solution very close to that which actually minimizes the probability of error. Since minimization of mean square error, or some other convenient measure of error such as alpha flunk, is required in a practical equalizer, the fact that the resulting solution is close to actually minimizing the error probability is reassuring.

REFERENCES

1. A. J. Viterbi, "Convolutional Codes and Their Performance in Communication Systems", IEEE Trans. on Comm. Tech., Vol. COM-19, October 1971, pp. 751-772.

2. J. A. Heller and I. M. Jacobs, "Viterbi Decoding for Satellite and Space Communication", IEEE Trans. on Comm. Tech., Vol. COM-19, October 1971, pp. 835-848.

3. K. J. Larsen, "Short Convolutional Codes With Maximal Free Distance for Rates 1/2, 1/3, and 1/4", IEEE Trans. on IT, Vol. IT-19, May 1973, pp. 371-372.

4. L. R. Bahl and F. Jelinek, "Rate 1/2 Convolutional Codes With Complementary Generators", IEEE Trans. on Info. Theory, Vol. IT-17, Nov. 1971, pp. 718-727.

5. E. Paaske, "Short Binary Convolutional Codes With Maximal Free Distance for Rates 2/3 and 3/4", IEEE Trans. on Info. Theory, Vol. IT-20, September 1974, pp. 683-689.

6. C. L. May and C. R. Cahn, "Adaptive Filter Functioning as a Channel Equalizer and Data Detector", Magnavox Technical Memorandum, MX-TM-3189-75, December 17, 1975.

7. C. R. Ryan and J. H. Stillwell, "Performance of a 5 Section Adaptive Equalizer for 350 Mbps QPSK Modem", EASCON, October 1974.

8. R. E. Ziemer and C. R. Ryan, "Equalization of QPSK Data Transmission in Specular Multipath", IEEE Trans. on AES, Vol. AES-10, September 1974, pp. 588-594.

9. J. H. Stillwell and C. R. Ryan, "Performance of a High Data Rate Adaptive QPSK Modem Under Media Distortions", Int. Comm. Conf., June 1975.

10. J. J. Jones, "Filter Distortion and Intersymbol Interference Effects on PSK Signals", IEEE Trans. on Comm. Tech., Vol. COM-19, April 1971, pp. 120-132.

11. J. W. Smith, "The Joint Optimization of Transmitted Signal and Receiving Filter for Data Transmission Systems", Bell Syst. Tech. Jour., Vol. 44, December 1965, pp. 2362-2392.

12. I. Korn, "Probability of Error in Binary Communication Systems With Casual Band-Limiting Filters - Part I: Nonreturn-to-Zero Signal", IEEE Trans. on Comm., Vol. COM-21, August 1973, pp. 878-890.

13. C. R. Cahn and R. S. Cnossen, "TDRSS Telecommunications Study Phase I - Final Report", Magnavox, Contract NAS5-20047, September 15, 1974.

14. Lucky, Salz, and Weldon, "Principles of Data Communication", McGraw-Hill, 1968, pp. 83-92.

15. G. D. Forney, "Maximum-Likelihood Sequence Estimation of Digital Sequences in the Presence of Intersymbol Interference", IEEE Trans. on Info. Theory, Vol. IT-18, May 1972, pp. 363-378.

16. M. J. Ferguson, "Optimal Reception for Binary Partial Response Channels", BSTJ, Vol. 51, February 1972, pp. 493-505.

17. A. L. Berman and C. E. Mahle, "Nonlinear Phase Shift in Traveling-Wave Tubes as Applied to Multiple Access Communication Satellites", IEEE Trans. on Comm. Tech., Vol. COM-18, February 1970, pp. 37-47.

18. C. R. Cahn, "Calculation of Intermodulation Due to Amplitude Limiting of Multiple Carriers", IEEE Trans. on Comm. Tech., Vol. COM-17, December 1969, pp. 743-745.

19. A. R. Kaye, et al, "Analysis and Compensation of Bandpass Nonlinearities for Communications", IEEE Trans. on Comm., Vol. COM-20, October 1972, pp. 965-972.

20. N. M. Blachman, "Detectors, Bandpass Nonlinearities, and Their Optimization: Inversion of the Chebyshev Transform", IEEE Trans. on Info. Theory, Vol. IT-17, July 1971, pp. 398-404.

21. K. G. Johannsen, et al, "Signal-to-Intermodulation Noise in Biased Limited Amplifiers", IEEE Trans. on Comm., Vol. COM-23, July 1975, pp. 743-757.

22. H. Kaufman and G. Roberts, "Correlation Function Expansion at the Output of a Nonlinear Device", Electronic Engineering, October 1963.

23. N. M. Blachman, "The Signal x Signal, Noise x Noise, and Signal x Noise Output of a Nonlinearity", IEEE Trans. on Info. Theory, Vol. IT-14, January 1968.

24. B. Widrow, et al, "Adaptive Antenna Systems, Proc. IEEE, Vol. 55, December 1967, pp. 2143-2159.

25. P. Butler and A. Cantoni, "Noniterative Automatic Equalization", IEEE Trans. on Comm., Vol. COM-23, June 1975, pp. 621-733.

26. B. Widrow, "Adaptive Filters I: Fundamentals", Stanford University, Technical Report No. 6764-6, December 1966.

27. C. L. May, "Equalization of PSK Channels Having Phase Distortion", Magnavox MX-TM-3160-74, December 11, 1974.

28. S. Zohar, "The Solution of a Toeplitz Set of Linear Equations", J. of Assoc. for Comp. Mach., Vol. 21, April 1974, pp. 272-276.

DATE
FILME